UNIVERSIDADE FEDERAL
DE SANTA CATARINA
PROGRAMA DE PÓS-GRADUAÇÃO
EM CIÊNCIA DA COMPUTAÇÃO

João Luiz Francalacci Rocha
www.inf.ufsc.br/~jrocha

Proteção de Software por Certificação Digital

Dissertação submetida à Universidade Federal de Santa Catarina como parte dos requisitos para a obtenção do grau de mestre em Ciência da Computação.

Prof. Ricardo Felipe Custódio, Dr.

Florianópolis, dezembro de 2001

Proteção de Software por Certificação Digital

João Luiz Francalacci Rocha

Esta Dissertação foi julgada adequada para a obtenção do título de mestre em Ciência da Computação Sistemas de Computação e aprovada em sua forma final pelo Programa de Pós-Graduação em Ciência da Computação.

Prof. Ricardo Felipe Custódio, Dr.

Prof. Fernando Gauthier, Dr.

Banca Examinadora

Prof. Ricardo Felipe Custódio, Dr.

Prof. Carlos Roberto de Rolt, Dr.

Prof. Luiz Carlos Zancanella, Dr.

Prof. Paulo Lício de Geus, Dr.

"There is a crack, a crack in everything.
That's how the light gets in..."
(Fjalar Ravia, "Fravia")

Para minha esposa Andréa e meu filho Lucas, que em muitas ocasiões durante o período de mestrado, foram privados da atenção e do carinho, conseqüência do empenho dedicado a este Projeto.

Agradecimentos

Gostaria de agradecer as seguintes entidades pela contribuição a este trabalho. São elas:

Entrevistados: todas as pessoas e empresas que contribuíram para a realização do estudo de campo;

LabSEC: ao LabSEC, na pessoa de Carlos Eduardo Silva e Luciano Ignaczack que ajudaram na emissão dos certificados do LabSEC para teste;

Augusto Jun Devegili: pelas dicas no LaTeX;

Banca: aos membros que comporam a banca examinadora desta Dissertação;

UFSC: a Universidade Federal de Santa Catarina e ao Departamento de Informática e Estatística, pelo excelente curso de pós-graduação e pela oportunidade de engrandecer meus conhecimentos;

Ricardo Felipe Custódio: um agradecimento em especial ao orientador deste Projeto, pela sua dedicação, sabedoria e paciência.

Conteúdo

Lista de Figuras

Resumo

A tecnologia quebrou fronteiras, criou supercomputadores, redes mundiais e software de última geração. Em contrapartida, surgiram problemas que variam desde a invasão de privacidade, falta de segurança dos dados até o desrespeito ao direito autoral. Este, talvez, um dos mais difíceis problemas que a tecnologia já lidou. Parece ser um vírus sem cura, um labirinto sem saída. Atualmente, a pirataria lesa milhares de produtores de software em todo o mundo, causa um prejuízo para empresas e fabricantes na ordem de bilhões de dólares e nada do que tenha sido feito até agora, nem de longe, teve uma eficácia real, nem mesmo as leis federais. O problema vai além dos chips dos computadores, passeia pela consciência do usuário, decola na facilidade de propagação da Internet e aterrissa na não degradação do meio digital.

Este trabalho é uma proposta para um combate mais eficaz contra cópias não autorizadas de programas de computador. É um estudo sobre infra-estrutura de chave pública com base na recomendação internacional X.509v3 e sua possível aplicabilidade para proteção de software usando a certificação digital. O propósito é condicionar o registro do software ao certificado digital do usuário. Se este mesmo usuário quiser fazer uma cópia pirata e distribuí-la, necessitaria fornecer, junto com a cópia, o seu certificado e sua chave privada, o que poderia trazer inúmeras complicações para ele, pois o certificado digital está associado ao usuário por meio de um contrato e esta associação não pode ser negada.

Este modelo oferece também, vantagens ao produtor/revendedor de software como: a) revogação do certificado que ativa as cópias ilegais; b) baseado no não repúdio mencionado acima, o produtor/revendedor pode processar os usuários que quebraram o termo de contrato; c) personaliza o software baseado nas informações contidas no certificado. Além disto, a validação do certificado é realizada pelo sistema operacional, não pelo software protegido. Desta maneira, se o usuário for sofisticado, este precisaria de tempo e esforço razoável para burlar o modelo de proteção.

Palavras-chave: pirataria, direitos autorais, ICP, certificação digital.

Abstract

The technology has broken boundaries, it has create supercomputers, the World Wide Web and the last generation of software. On the other hand, many problems have been noticed: invasion of privacy, insecure data storage and disrespect to the copyright. This perhaps, one of the most difficult problems that the technology has ever faced. It seems to be a virus without cure, a maze without exit. Currently, the piracy causes a global revenue losses in such magnitude of billions of dollars and nothing that was made could efficiently stopped it, even the federal laws. The problem of software piracy goes beyond the computer chip, walks through the user's conscience, takes off on the easy Internet propagation and it lands in the non degradation of digital media.

This paper is a proposal to fight against unauthorized copying of computer software in a more efficient way. It is a study about code signing supporting the international X.509v3 recommendation and how it could be applied for copy protection using digital certification. The purpose is to set conditions for software registration by digital certification which means that the user must inform his/her digital certificate. After the registration done, if the user wants to make illegal copies of the software he/she would have also to distribute his/her digital certificate and private key. Unlikely, once it would bring undesirable and legal consequences because the certificate is associated to the user by a contract and this association can not be denied.

In addition, this model offers advantages to the software provider such as: a) the provider can revoke the certificate that activates illegal copies; b) based on the non repudiation mentioned above, the provider can sue the users which broke the term of contract; c) it personalizes the software based on the certificate contents. Also, the certificate validation is carried out by the operating system, not by the copy protected. So, if the user is sophisticated he/she would need to spend reasonable time and effort to crack the protection.

Keywords: piracy, copyright protection, PKI, digital certification.

Capítulo 1

Introdução

Este documento tem como objetivo o desenvolvimento de um modelo mais eficaz de combate à pirataria. Diferentemente de outras formas de combate existentes, este novo modelo oferece vantagens ao produtor/revendedor de software com base nas seguintes propriedades:

- maior controle sobre as cópias piratas existentes no mercado, tendo o poder de neutralizá-las através da revogação do certificado de licença de software;
- proporcionar meios eficazes de provar judicialmente a autoria do crime cometido, ou seja, a cópia ilegal, através da ligação entre o usuário infrator e o certificado de licença de software;
- determinar concessões de direito de uso do software, previstas em contrato e estabelecias pelo período de validade do certificado de licença de software;
- proporcionar a personalização do software com base nas informações contidas no certificado de licença de software.

Estas propriedades, por não pertencerem a nenhum tipo de proteção de software na atualidade, são principal motivação desta pesquisa. Elas são metas a serem atingidas até a conclusão desta dissertação, através da pesquisa para o desenvolvimento de um modelo teórico de funcionamento e sua aplicação na prática

por meio da implantação do modelo em um protótipo. Para que isto se torne viável, estudos na área de segurança em computadores, como assinatura digital e infra-estrutura de chave pública são necessários, bem como a emissão de certificados digitais para testes do protótipo.

Esta obra é uma dissertação de mestrado do CPGCC (Curso de Pós-Graduação em Ciência da Computação) da Universidade Federal de Santa Catarina e abrange a área de segurança e comércio eletrônico. Ela começa com o capítulo 2, onde, primeiramente, falamos sobre pirataria de software, quais mecanismos de controle estão sendo empregados atualmente, verdades e mentiras, as tendências da pirataria e o desrespeito à propriedade intelectual, com o intuito de provar que, mais do que nunca, se faz necessário investir em pesquisa para criação de uma forma eficiente de combate à pirataria. Em seguida, no capítulo 3, abordamos o assunto sobre assinatura de código, na tentativa de mostrar o que existe hoje. Este assunto é de grande importância para o entendimento de como é realizada a assinatura digital e de como é possível garantir-se a integridade e autenticidade de um código. Depois, no capítulo 4, um estudo sobre infra-estrutura de chave pública será feito com o objetivo de mostrar ao leitor a estrutura existente que viabiliza todo o processo de certificação digital, envolvendo recomendações, caminhos de certificação e autoridades certificadoras.

No capítulo 5, apresentamos uma proposta com o objetivo de demonstrar com mais detalhes a maneira como planejamos desenvolver e implementar a Proteção de Software por Certificação Digital e, por último, no capítulo 6, falamos sobre as considerações finais e concluimos esta obra.

O apêndice traz, a título de curiosidade, um ensaio preliminar de estudo de campo com alguns resultados obtidos e os questinonários que foram aplicados.

Capítulo 2

O Controle, a Pirataria de Software e a Propriedade Intelectual

2.1 Introdução

Nas últimas décadas a evolução no campo da informática foi imane. A tecnologia permitiu aos computadores invadir as fronteiras domésticas de tal maneira que o microcomputador ou "computador pessoal" tornou-se um eletrodoméstico freqüente nos lares de classe média/alta da sociedade. A Internet tornou-se realidade para milhões de pessoas no mundo inteiro, de fácil acesso e com uma fonte praticamente inesgotável de informações. Este fenômeno tecnológico popularizou milhares de programas para computador e os fez cada vez mais necessários no dia a dia do usuário. Contudo, os usuários logo perceberam que a imensa maioria dos programas podia ser obtida sem que fossem necessariamente comprados. Bastava ter o original do programa para se desencadear uma verdadeira disseminação de cópias piratas. Nascia, então, a indústria da pirataria, uma atividade ilegal exercida intensamente em todo o planeta.

Mas por qual razão a pirataria é tão praticada? Começaremos este

capítulo, falando, no item 2.2, sobre a formação de uma cultura negativa para nossa sociedade como consequência da prática da pirataria. Depois, no item 2.3, abordaremos algumas formas mais comuns de combate à pirataria. Não podemos deixar de falar sobre o lado bom da pirataria. Veremos isto no item 2.4. No item 2.5, expomos as razões que levam as pessoas a exercerem esse ato ilegal e que causa enormes prejuízos às indústrias de software e a sociedade em geral, pois se as empresas não conseguem vender seus produtos, parte da economia sofre uma recessão, empregos deixam de ser criados e o governo deixa de arrecadar impostos. Abordaremos também, no item 2.6, o panorama atual da prática de cópias ilegais no Brasil. No item 2.7, faremos uma conclusão deste capítulo.

2.2 A Cultura Cega

Igualmente a evolução da informática, a pirataria de software e o desrespeito à propriedade intelectual subiram em proporções assustadoras. Hoje é muito comum adquirir cds [1] piratas no "camelódromo" da cidade, efetuar download de programas "crackeados" pela Internet e até mesmo encontrar anúncios de cds piratas nos jornais - como mostra a figura 2.1. A facilidade e a impunidade são tão grandes que estamos correndo o risco de criar uma cultura negativa irreversível (ou no mínimo difícil de se reverter) nos milhões de usuários de computadores. O pior é que cultura, seja ela boa ou má, passa de geração para geração. Este assunto é bem citado no relatório Global Software Piracy 2000 do SIIA [2] - *"...a facilidade de duplicação e a alta qualidade do software pirateado, representa um problema significante para a indústria do software"* [ASS 00], isto porque não há uma degradação na qualidade do software quando ele é copiado e recopiado, pois diferentemente da gravação analógica, a gravação digital não sofre perda no processo de cópia. Os fatores que mais contribuem para o aumento desta contracultura são a ganância, falta de cuidado e ignorância para com as leis vigentes e falta de respeito

[1] Abreviatura de "compact disk" no plural.

[2] SIIA – Software & Information Industry Association [ASS 00]

PROGRAMAS 99/2000
EM 5 CDS R$ 59,00

Coletania com garantia e suporte tec. grátis contendo: Win 98BR, Office BR Word, Acess Excell, Corel Draw BR, Dic. Aurélio e Michaelis BR, Front Page98, Print Artist 4, Translator PRO 6.4, Page Maker 6.5 BR, Empresario 2BR, PhotoShop 5.0, Norton Antivirus/Utilities/Internet/CrashGuard BR, Director 7, Money99, Gaspro 99, Premier 5.0, TurboCad, CleansWeeper 4 BR, Freehand 8.1 Phone 5.0, Winzip 7, Visio 5, OminiPage PRO 9, Netscape 4.5, Winfax 9, VirusCan 3.1BR, HOT Dog 5.0, GIF Animator 2.0, C++Builder, Hollywood FX 3.0, I Explorer 5.0, N Internet 1.0, QEMM, J Builder, Revisor Gramatical 3.0, DTS, Play-Boy Quicken 99, +35 Super Progms completos. Temos Delphi 4, VB6, SAP 2000, Eberic, Eng. total, Juridicos, Cy-Pecad 98 e Medicina, Win NT, Curso AutoCAD R-14, Jogos Lançamentos, Banco Imobiliário, Sincity 3000, moto Race II, Dune 2000. Lançamentos a R$ 25,00 cada CD Windows 2000 - Office 2000 - Norton 2000 - AutoCad 2000 - FrontPage 2000 - Novel 5.0 - Barsa 99 - Encarta 99 - Almanaque 99 - Corel Draw 9.0. Entrega para Todo Brasil em 24 hs

Ligue grátis fones: 9(41) 915-8090 e 9 (041) 915-7090

30, 18 DE JULHO DE 1999

Figura 2.1: Publicado no Diário Catarinense, 18 de julho de 1999. Este anúncio foi publicado em importante jornal de SC na época do lançamento do Windows® e Office® 2000. Quando indagado sobre a procedência do software, o anunciante admitiu ser produto pirata e tentou amenizar a questão alegando oferecer suporte e documentação eletrônica. Note que o anunciante aceita chamadas a cobrar para as encomendas.

para com a propriedade intelectual [ASS 00] e [WIN 99]. As pessoas que copiam ilegalmente software, seja o criminoso que copia em larga escala, seja o simples usuário que empresta seu software legal a um amigo ou a empresa que compra uma única licença e instala o software em diversos computadores, todos praticam a pirataria com a desculpa de que "todo mundo faz isto". Este sentimento errôneo de abstenção de culpa incentiva uma pirataria que, segundo o relatório Global Software Piracy 2000 do SIIA, nos últimos três anos tem crescido muito, principalmente nos países que não possuem mecanismos legais de combate a prática deste crime. Só na América Latina este prejuízo saltou de US$970 milhões em 1997 para US$1,128 bilhão em 1999. Este valor da América Latina, comparado aos prejuízos no resto

do mundo, representa pouco menos de 10%, pois a soma chega a US$12,163 bilhões. Estes números não representam totalmente a realidade, uma vez que as pesquisas levam em conta que para cada software pirateado, necessariamente uma compra do software legal deveria ser feita, o que não é verdade, pois nem sempre o usuário compra o software que não consegue piratear.

2.3 As Tentativas Atuais para Controlar a Pirataria

A indústria do software não está passiva ao problema da pirataria. Pelo contrário, muitas tentativas de proteger o software já foram testadas. O problema é que este assunto é delicado para as empresas que produzem software. As empresas escondem ou procuram não divulgar isto aos seus clientes, pois proteção de cópia pode ser um motivo de queda nas vendas já que o software torna-se difícil de ser copiado, impedindo que o usuário faça uma cópia de segurança e podendo, assim, se tornar uma futura dor de cabeça para o comprador. Esta característica, a de tornar difícil um backup de segurança do software mais a dependência com dispositivos de hardware para que o programa funcione, contribuem para que as proteções de cópia existentes hoje não sejam bem sucedidas [CHO 95].

Além disso, existe o aspecto ético, por vezes esquecido. Muitas empresas, na pressa do domínio de mercado, lançam software com problemas ou mal acabados que no final das contas, prejudicam o consumidor. Este, depois de ter comprado um software com defeito, nada pode fazer, a não ser, esperar pelo lançamento de um pacote reparador do fabricante do software. Se o defeito for causado por algum esquema de proteção, então o problema se agrava, pois a empresa, na tentativa de se proteger dos piratas, prejudica seu próprio cliente: *"Acontece que as empresas de software proprietário protegem com todas as armas seu modelo de negócio, não admitindo inseri-lo em nenhum contexto social que não seja o do sua contabilidade. Seus sistemas dificilmente teriam os intestinos publicamente dissecados. Pois seu*

código-fonte, a versão em linguagem semi-humana na qual são projetados e construídos, não estará disponível aos licenciados. E a engenharia reversa dessas caixas pretas, o equivalente digital da autópsia, é proibido e severamente criminalizado pelas leis de proteção ao direito industrial do software, geralmente conhecidas como leis anti-pirataria, promulgadas sob intenso lobby dessas empresas." , [dR 01].

2.3.1 Um Modelo Genérico de Pagamento Eletrônico com Suporte a Múltiplas Transações Comerciais

Em 1999 surgiu a proposta para implantação de um modelo genérico de pagamento eletrônico com suporte a múltiplas transações comerciais [YH 99], o qual foi desenvolvido no Departamento de Engenharia da Informação e Ciência da Computação da Universidade Nacional Chiao Tung situada na Tailândia. Participaram do projeto Fu-Shen Ho, Yu-Lun Huang e Shiuh-Ping Shieh, sendo que este último é diretor da Computer and Network Center e professor da citada Universidade.

O projeto é um modelo que propõe atender transações eletrônicas para múltiplos comerciantes, onde varejistas, revendedores, provedores de conteúdo estão envolvidos nestas operações de distribuição de conteúdo eletrônico. Como conteúdo eletrônico, entende-se qualquer arquivo digital como: documentos, música, vídeos, programas, etc. O ponto forte do modelo encontra-se no retorno garantido dos proventos relativos ao direito autoral. Ou seja, desde a saída do conteúdo eletrônico do provedor de conteúdo até a chegada do produto ao consumidor; o detentor dos direitos autorais receberia sua parcela, bem como todas as entidades envolvidas na operação de venda (provedor de conteúdo, comerciante, AC e financiadora). Para que isto seja garantido, o modelo faz uso da criptografia simétrica e assinatura digital para distribuição da mídia. As licenças de vendas tratadas por: provedores de conteúdo, comerciantes e financiadoras; e os recibos digitais tratados por: comerciantes, consumidores e financiadoras, são documentos legais e garantidos por uma AC que asseguram a natureza da operação e afastam a possibilidade

de repúdio de compra ou venda entre as entidades envolvidas, uma vez que, tanto licença como recibo para terem validade precisam ser assinados por ambas as partes e autenticados por uma AC.

Apesar do modelo garantir que pode desencorajar a prataria, na verdade ele só evitaria a pirataria praticada entre os comerciantes intermediários (que realmente acontece [YH 99]), pois somente o consumidor poderia decifrar o conteúdo eletrônico. Porém, uma vez decifrado o conteúdo, os arquivos que estavam encapsulados podem ser facilmente pirateados.

Finalizando, os autores propõem a implementação do modelo em dois protocolos de pagamentos já conhecidos: o SET [3] e o NetBill [4].

2.3.2 Proteção de Cópia para Publicação Eletrônica em Redes de Computadores

Os autores A. K. Choudhury, N. F. Maxemchuk, S. Paul e H. G. Shulzrinne, reforçam a idéia de que uma das maiores dificuldades da tecnologia atual é evitar a pirataria de documentos eletrônicos. Citam ainda que a distribuição de documentos eletrônicos é muito mais rápida, mais barata e requer menos esforço do que efetuar fotocópias de documentos da maneira tradicional.

O que eles propõem para combate à pirataria é uma arquitetura com dois esquemas distintos para possibilitar uma distribuição de documentos eletrônicos segura [CHO 95]. A primeira estratégia e a mais segura requer que os periféricos (vídeos e impressoras) sejam comercializados já com um "firmware" específico para suporte à criptografia utilizada pelo modelo. A segunda estratégia, mais imediata e viável do ponto de vista econômico, requer que um software seja instalado no computador do usuário. Esta segunda estratégia, porém, é vulnerável, sob forte empenho, a ataques de engenharia reversa, onde o usuário sofisticado com recursos

[3] Secure Electronic Transaction é um famoso protocolo de pagamento proposto pela VISA e pela MasterCard

[4] Da autoria de B. Cox, J. D. Tygar e M. Sirbu, é um sistema para pagamentos de mercadorias vendidas através da Internet

e bastante paciência, poderia alterar as chamadas de autenticação do sistema e derrubar o esquema de proteção. Ambas as estratégias fazem uso da criptografia.

A segunda estratégia onde é necessária a instalação de um software na máquina do usuário exige que seja usado um par de chaves do usuário requerente do documento, onde a chave privada dele será usada para decifrar a chave secreta e, conseqüentemente, o documento. Igualmente, como foi declarado neste Projeto, os autores também acreditam que o usuário final ficaria desencorajado a produzir cópias ilegais uma vez que teria que distribuir junto com as cópias ilegais, cópias da sua chave privada.

2.3.3 Outras Formas de Proteção de Cópia de Software

A proteção de cópia pode vir de diversas formas, é o que diz Ethan Winer em seu artigo "The Audio Industry´s Dirty Little Secret":

Número de série ("cd-key"): *"A forma mais simples de proteção de cópia requer que você entre com um número de série no ato de instalação do programa. Na prática isto protege muito pouco, já que qualquer um pode emprestar os discos de instalação e o número de série para um amigo"* [WIN 99]. Esta forma também é a que menos dor de cabeça causa ao usuário, uma vez que basta ele não perder o número de série para efetuar futuras instalações;

Disquete de proteção: Uma outra forma de proteção seria alienar o software com um disquete de instalação teoricamente "incopiável" e requerido pelo software a cada nova instalação. Esta forma já apresenta problemas, visto que o disquete pode apresentar defeitos e causar aborrecimentos ao usuário. Como exemplo deste tipo de proteção podemos citar o HandProt [5];

Contra-senha: A forma de proteção mais comum atualmente obriga o usuário a contatar a empresa produtora do software via telefone ou e-mail para obter uma contra-senha e habilitar todas as funções do programa;

[5]Produto comercializado pela Squadra:www.squadra.com.br

Dispositivo de hardware ("dinkey" ou "hardware-key"): A pior forma (para o usuário legal) de proteção - *"O método mais punitivo de proteção de software usa um dispositivo de hardware que vem com o produto e necessita ser conectado na porta USB ou paralela do computador. Se o dispositivo não estiver conectado, o programa detecta isto e não funciona"* [WIN 99]. Como exemplo de empresas que utilizam esta tecnologia podemos citar a Microcosm (www.microcosm.co.uk) e a Griffin Technologies (www.griftech.com).

Nenhuma dessas formas de proteção possui uma eficácia total. É comum encontrarmos versões de programas "crackeados" pela Internet ou em cds piratas. Os "crackers" fazem uso de decompiladores e da engenharia reversa para "retirar" as proteções implantadas no programa e lançar as versões "crackeadas".

Apesar das formas de proteção contra cópia acima descritas terem surgido ao longo dos anos, nenhuma delas conseguiu romper a barreira da dependência usuário/fabricante causada pelos métodos aplicados. O usuário sempre acaba prejudicado na troca ou falha de equipamentos, pois na reinstalação do software adquirido, normalmente, ele precisa contatar o fabricante novamente para prosseguir na reinstalação. A situação fica crítica se o fabricante não existe mais ou não fornece mais suporte para a versão do software. Este quadro tende a piorar, pois com a baixa dos preços dos equipamentos, as atualizações de hardware, como troca de winchester e mesmo troca completa do computador, irão ocorrer com cada vez mais freqüência.

2.4 O Lado Positivo da Pirataria

Algo está errado. Será que os métodos de proteção de cópia atuais estão funcionando? Será que estes métodos estão ajudando os fabricantes de software a elevar as vendas? O usuário certamente ficaria com um "pé atrás" quando fosse comprar um software que possua uma proteção de cópia tão punitiva. As vendas de podem ser prejudicadas pelo simples fato de usarem proteções de cópias, sendo elas

transparentes ou não, pois ninguém gosta de ficar na dependência de terceiros ou até mesmo de ficar impossibilitado de fazer um "backup" de segurança do software.

Por esta razão, muitos fabricantes de software optam por uma proteção sutil, como apenas um número de série ou mesmo proteção alguma. Claro que a possibilidade de pirataria é muito maior, mas há depoimentos de empresários como o de Bob Lentini da Innovative Quality Software, que não vêem tanto perigo na pirataria: *"Nós não usamos proteção de cópia em nenhum de nossos produtos atuais. Tenho tido uma longa experiência de que as versões piratas não causam tanto prejuízo como se pensa. Os "crackers" nunca pensariam em comprar um produto se eles não pudessem roubá-lo. Muito de nossos clientes vieram até nós depois de usar uma versão pirata por um período breve e depois eles decidiram que não poderiam mais viver sem o produto e fizeram o registro para ter acesso a suporte, "download" gratuito e outras vantagens. Outros compraram após verem uma versão funcionando em outro computador...pura propaganda"* [WIN 99].

Para complementar, é importante dizer que o maior "marketing" proporcionado pela pirataria vem da propaganda de boca em boca. Quanto mais se alastra o número de cópias piratas de um determinado software, maior será a fatia de mercado em potencial para o fabricante deste software. Muitas empresas fazem uso da pirataria como estratégia para ganhar mercado. Afinal, treinamento é um dos fatores que mais influenciam em compra de software, pois exige tempo e dinheiro. Um software bem aceito pela comunidade de usuários significa um software de conhecimento global, o que elimina, em muitas vezes, a necessidade de treinamento.

2.5 O Que Incentiva Tanta Pirataria

Atualmente, não há proteção de cópia forte o suficiente que resista a ação dos "crackers". Quanto mais forte for a proteção, maior é o incentivo intelectual para se quebrar o esquema. Hoje, é bastante comum encontrar fóruns de discussão sobre como piratear programas, sites com dicas e software "crackeados" para download na Internet. A +HCU: Academy of Reverse Engineering, como

é assim denominada pelos especialistas da área de Engenharia Reversa, em seu site http://www.instinct.org, é um exemplo típico de como se obter informações para se quebrar as proteções de cópia existentes no mercado. Os especialistas nesta área tratam até de forma irônica a tentativa dos programadores em proteger seus produtos: *"...Pobres programadores de shareware...eles gostariam tanto de se concentrar em seus próprios códigos bem feitinhos, mas -ai deles!- eles têm que travar batalhas contra uma multidão de piratas, milhares de aficionados em "serial numbers", a própria concorrência deles (o que deve ser o maior dos inimigos), e por último, mas não por menos, todos os "crackers" na rede mundial de computadores (eu não incluo os especialistas em Engenharia Reversa nesta lista porque nós não somos os inimigos deles e também porque a luta deles contra nós teria a mesma chance de um papel contra uma tesoura :-) Alguns deles no mais profundo desespero (e inacreditável ignorância de assembler) resolvem comprar um esquema comercial de proteção de cópia já pronto, muitos dos quais não funcionam de maneira alguma..."* [oRE 97].

Devemos dar crédito as palavras de Ethan Winer em seu artigo, para uma explicação ao grande incentivo da prática da pirataria: *"Pessoalmente, eu acredito que o problema real é que o software é exageradamente caro. As pessoas querem fazer a coisa certa e estarão dispostos a pagar por um programa que atenda as necessidades deles se eles tiverem condições de pagar o preço"* [WIN 99]. Mas, claro, não é só isto. Como foi citado antes, a ignorância, o descaso, o fácil acesso ao produto pirateado, etc..., são outros fatores que contribuem para o incentivo à pirataria. Poderíamos, então, classificar estes fatores:

- o alto custo do software;
- a falta de leis e política de proteção autoral e conseqüente impunidade para os infratores;
- a falta de uma política educativa;
- a facilidade de acesso e a não degradação dos produtos piratas (Internet, cds, disquetes $3\frac{1}{2}$ pol., zip drives, etc...);

- a inexistência de uma proteção eficaz para parar este processo.

2.6 A Pirataria no Brasil

Em termos de América Latina, o Brasil ocupa um lugar de destaque no campo da pirataria. Junto com a Argentina e México, chegamos a 2/3 das perdas provocadas pela pirataria na América Latina. No Brasil, o panorama era extremamente grave até 1987. Depois de sancionar a Lei de Software, lei número 7.646, em 18 de dezembro de 1987, o Brasil passou a se incluir entre os países que possuem legislação específica de proteção à indústria do software. Após a introdução desta lei, ficou estabelecido que a violação dos direitos autorais de programas de computador é passível de ação cível de indenização.

Assim como nos Estados Unidos e França, onde a Lei de Software é casada com o Tratado Mundial de Propriedade Intelectual - WIPO Copyright Treaty (World Intellectual Property Organization), no Brasil a Lei de Software também é combinada com a Lei do Direito Autoral, lei número 5988/73. A Lei do Software estabelece que as perdas e danos do titular do programa sejam ressarcidos pelo valor equivalente a 2.000 cópias de cada software ilegalmente reproduzido. O infrator fica sujeito também à retenção de 6 meses a 2 anos, além de multas diárias pelo uso ilegal do software pirata. A partir de 1989 a ABES (Associação Brasileira de Empresas de Software) iniciou uma campanha antipirataria no Brasil e três anos depois a Business Software Alliance, uma entidade dos Estados Unidos que reúne os principais produtores de software em nível mundial, uniu forças com a ABES para combater a pirataria, promovendo ações de busca e apreensão em todo o país. Porém, mesmo depois da lei 7.646 ser sancionada em 87 e das iniciativas da ABES, o índice da pirataria ainda continuou alto. Faltava uma maior conscientização da sociedade e faltava melhorar a lei para torná-la mais completa, favorecendo e impondo punições, tanto para os fabricantes de software como para os consumidores. Foi, então, que o Congresso Nacional decretou a nova Lei do Software, lei número 9.609 de 19 de fevereiro de 1998 [BRA 98], a qual trata com mais rigor a questão de proteção

aos direitos autorais e garantias ao usuário de programa de computador, inclusive elevando a pena de retenção de um para quatro anos. Apesar de todos os esforços o Brasil, assim como o resto do mundo, ainda continua amargurando prejuízos em conseqüência da pirataria.

2.7 Conclusão

Se fecharmos os olhos para o problema da pirataria, esta com certeza sairá do controle dos comerciantes e produtores de software, governo e associações de proteção. O que irá "banir" a pirataria não é apenas uma única medida, mas sim a união de vários esforços. Precisamos educar o público consumidor alvo, precisamos elaborar leis mais rigorosas, precisamos também saber aplicar estas leis com eficiência para punir infratores, precisamos baixar o custo do software e precisamos criar um dispositivo eficiente que quebre o ciclo vicioso da pirataria.

Capítulo 3

Assinatura de Código

3.1 Introdução

A Internet tornou-se uma verdadeira vitrine para demonstração de software. Os programas para download ganharam várias denominações. Entre as mais conhecidas, estão: "freeware", um tipo de programa oferecido gratuitamente e "shareware", "trial" ou "demo", que são versões de programas que funcionam durante certo período para avaliação do usuário. Inúmeros sites de download surgiram e os produtores de software que deixam de disponibilizar versões de demonstração de seus produtos na Internet perdem uma boa chance de aumentar suas vendas. Todo este progresso e facilidade trouxeram um problema significativo. Como confiar no programa que está sendo baixado? Será que realmente pertence ao produtor mencionado na página? Será que o software não está infectado com vírus?

Uma das primeiras notícias que se tem conhecimento sobre abalo na relação de confiança entre usuários e programas para download é datada de 1995, quando um programa chamado "pkzip30b.exe", produzido pela empresa Pkware para compactar arquivos em disco, foi propositadamente alterado por "hackers" para remover os arquivos do disco e depois disponibilizado em vários sites na Internet para download. O estrago causado na comunidade de usuários comprometeu significativamente a reputação da empresa criadora do software [GAR 99], pg.172.

Exatamente para resgatar essa relação de confiança, que a assinatura de código foi criada. Neste capítulo, estudaremos a assinatura de código, começando pela sua definição no item 3.2 e, em seguida, no item 3.3, os tipos de assinatura de código. No item 3.4, falaremos sobre o Authenticode. Vamos falar, também, sobre a função resumo no item 3.5 e alguns de seus algoritmos no item 3.6. Com o objetivo de proporcionar um entendimento maior sobre assinatura digital, veremos este tópico no item 3.7, onde explicaremos como ela é realizada e como é possível garantir a integridade e autenticidade de um código. Falaremos brevemente sobre a CryptoAPI no item 3.8 e, por fim, no item 3.9, concluiremos este capítulo.

3.2 Definição e Benefícios da Assinatura de Código

Podemos definir a assinatura de código como uma técnica utilizada para "lacrar" programas de computador através da assinatura digital.

A assinatura de código proporciona ao usuário a segurança de um software lacrado, o que quer dizer que ela assegura autenticidade e integridade. Por autenticidade, ao usuário fica garantido que o código realmente está assinado por quem diz ser o assinante, e por integridade, fica garantido de que o código não foi alterado após ter sido assinado. Além disso, se o software executar atividades maliciosas ou danosas no computador onde foi instalado, o usuário tem recursos legais para processar o assinante do código.

3.3 Tipos de Assinatura de Código

Podemos dizer que existe uma diferença crucial na relação de confiança entre usuários que compram software através de lojas para os que compram software através da Internet. Quando um usuário vai a uma loja e compra um determinado software, que vem em embalagem lacrada, com a fonte produtora visivelmente citada e com selo de garantia e autenticidade, a confiança é imane. Já os usuários que compram software disponível para download na Internet, naturalmente,

não compartilham do mesmo grau de confiança.

Atualmente, existem várias ferramentas e formas de assinatura de código. Pode-se citar como exemplo: o Netscape Object Signing, desenvolvido para os usuários do navegador da Netscape e o famoso programa para segurança de dados, PGP[1] (Pretty Good Privacy). Este último, não se consolidou como opção para assinatura de código na Internet, pelas seguintes razões: i) o fato de não haver suporte para o PGP nos navegadores da Internet, a validação da assinatura não pode ser realizada antes do download ser efetuado; ii) o resumo da assinatura não é anexado ao código e iii) O PGP não usa infra-estrutura de chave pública.

Ao contrário do PGP, o método que vem obtendo maior aceitação e êxito é o desenvolvido pela Microsoft®, o Authenticode. Este sim atende aos requisitos para baixa de códigos pela Internet, pois é suportado pelos navegadores, anexa a assinatura ao código e usa infra-estrutura de chave pública.

3.4 Authenticode

O Authenticode surgiu para criar essa confiança na comunidade de usuários virtuais que utilizam o Windows® e o Internet Explorer™, para fazer com que eles sintam-se seguros em baixar programas ou "plug-ins", garantindo a integridade e autenticidade do produto. Para tornar isto possível, o Authenticode conjugado com identidades digitais, utiliza a assinatura digital para assinar códigos e habilitar os produtores de software a incluir informações sobre eles em seus códigos.

A versão 2.0 do Authenticode é capaz de assinar executáveis de 32 bits e arquivos com extensão .cab, .ocx, .class, controles ActiveX, plug-ins e Java applets.

[1]PGP é um programa largamente difundido que provê, entre outras coisas, serviço de autenticação e confidência. Maiores informações sobre o PGP estão disponíveis em: http://www.nai.com.

3.4.1 Como funciona o Authenticode

Quando o código é assinado e disponibilizado na Internet, o usuário pode verificar ou ser avisado da existência ou não desta assinatura, antes mesmo de iniciar o download do código. A figura 3.1 ilustra um usuário baixando um código assinado pela Internet.

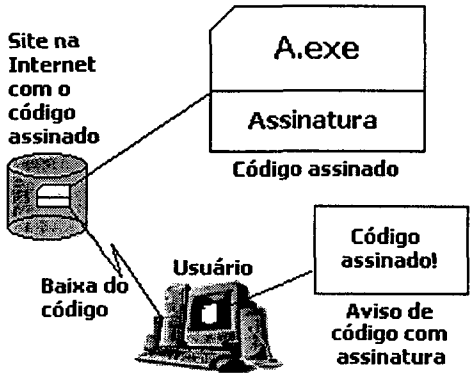

Figura 3.1: Baixa de código assinado pela Internet. O usuário baixa pela Internet um código "A" qualquer que está assinado e guardado em um servidor. Quando o processo de baixa se inicia, o usuário é avisado que aquele código está assinado e é um código em que ele pode confiar.

Vejamos agora um exemplo do uso de um navegador popular como o Internet Explorer (3.0 em diante). O Internet Explorer possui 3 níveis[2] de segurança que podem ser configurados pelo usuário. Se o nível de segurança do navegador for o máximo exigido, não permite que o usuário baixe código algum sem que esteja assinado. Porém, se o nível de segurança for médio, o usuário será alertado quando baixar qualquer código que não esteja assinado, conforme ilustra a figura 3.2. Assim, ele terá a chance de decidir ser quer ou não efetuar a operação.

Do mesmo modo, se o código se encontrar assinado, o usuário será informado da identidade do assinante daquele código, sobre onde encontrar mais informação a respeito do assinante e sobre qual autoridade certificadora garante estas informações. Isto é o que mostra a figura 3.3.

[2]Para obter maiores informações sobre os níveis de segurança, consulte o manual eletrônico do navegador Internet Explorer.

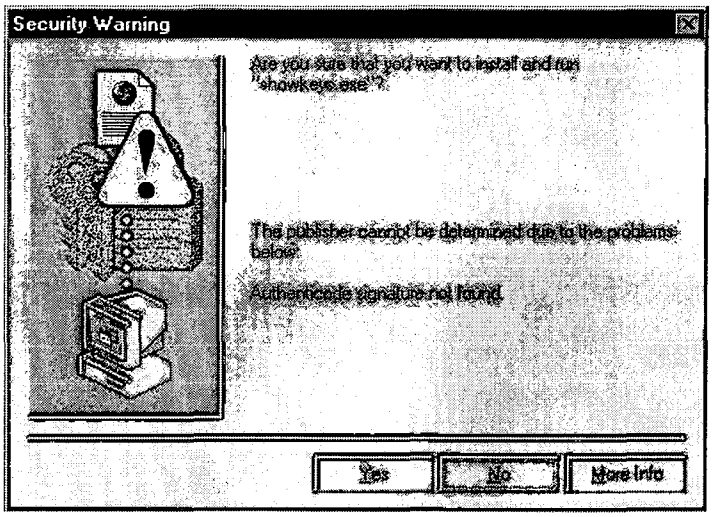

Figura 3.2: Aviso de ausência de assinatura. Esta figura ilustra o momento em que o Internet Explorer™ avisa ao usuário que ele vai iniciar o processo de baixa de um arquivo denominado "showkeys.exe", que não se encontra assinado.

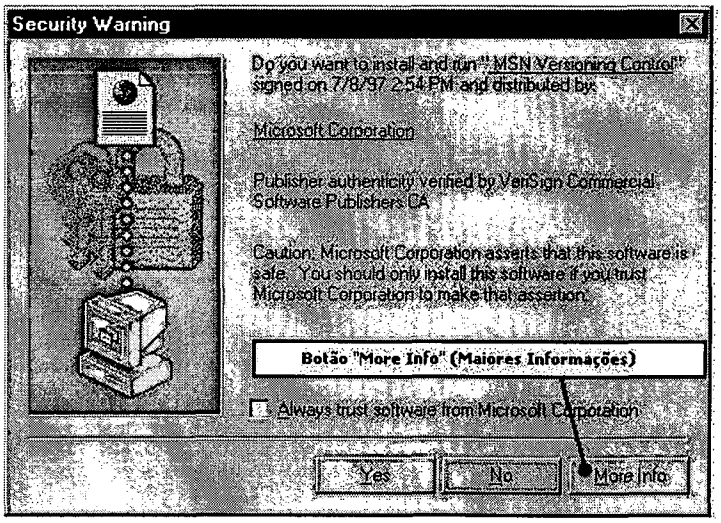

Figura 3.3: Aviso de código assinado. Neste caso, o usuário está sendo avisado que o código "MSN Versioning Control" é confiável, foi assinado em 08/07/97 e é distribuído pela Microsoft® Corporation. Se o usuário quiser maiores detalhes sobre o certificado que assina este código, ele deve clicar no botão "More Info".

Conforme a ilustração da figura 3.3, se o usuário clicar no botão "More Info" (Maiores Informações), a tela de propriedades do certificado irá apare-

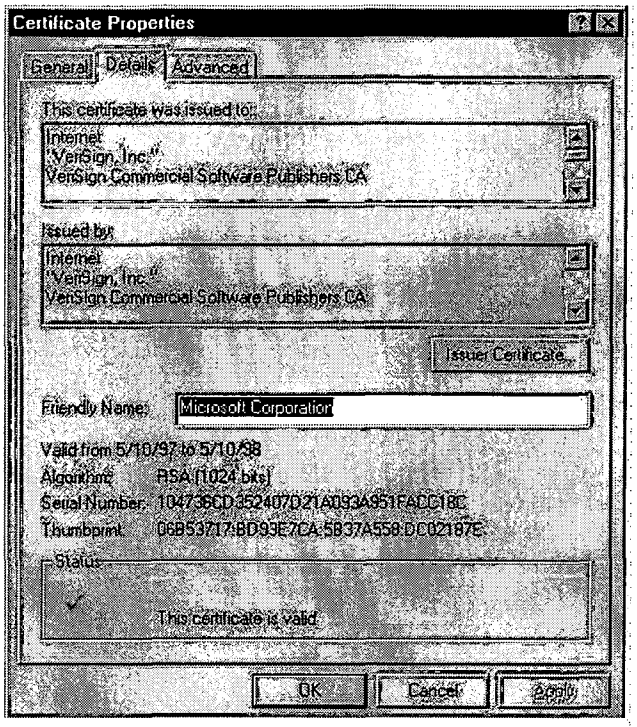

Figura 3.4: Propriedades do certificado. Esta figura ilustra detalhes importantes sobre o certificado usado para assinar o código da figura 3.3. Aqui podemos ver, entre as informações mais importantes, a autoridade certificadora que autentica o certificado do assinante, o período de validade do certificado, seu número de série e o status de que o certificado é válido.

cer, tornando possível verificar a validade do certificado. Isto é ilustrado na figura 3.4.

Quanto ao nível de segurança oferecido pelo Authenticode relacionado a força dos algoritmos usados pelo programa, a Microsoft garante: *"O Authenticode usa tecnologia de assinatura digital extremamente segura (1024 bits). É estimado que para se quebrar uma assinatura digital de 1024 bits, seriam necessários 90 bilhões de anos MIPS, ou seja, um bilhão de computadores executando um milhão de instruções por segundo durante 90 anos"* [COR 96b], pg.3. Com este nível de segurança, tentativas de engenharia reversa sobre a assinatura digital seriam extremamente difíceis de serem feitas com sucesso. A segurança desta tecnologia vem de recomendações de criptografia adotados internacionalmente como

certificados X.509v3 e infra-estruturas de chave pública. Estas recomendações são confiáveis e de eficácia comprovada.

O procedimento para a assinatura do código começa com a obtenção de um certificado digital destinado a desenvolvedor de software. Depois com a aplicação pronta, o desenvolvedor executa o utilitário signcode.exe [3] , informando um algoritmo de cálculo de resumo: MD5 ou SHA-1. O resumo é então cifrado com a chave privada do desenvolvedor e um pacote contendo o código, resumo cifrado e o certificado do desenvolvedor é gerado.

3.5 Função Resumo ou Função Hash

É uma função que gera um número de tamanho fixo e pequeno, que representa de forma única um documento eletrônico. O valor resumo é gerado por uma função r na forma:

$$r = H(Ob)$$

O (Ob) representa um objeto de tamanho variável e o $H(Ob)$ ou r, o valor resumo de tamanho fixo. Todas as funções resumo funcionam com uma entrada, arquivo, mensagem, etc..., que é vista como uma seqüência de blocos de n-bits processados um a um de maneira interativa para produzir a função resumo de m-bits.

3.5.1 Propriedades de uma Função Resumo

O propósito de uma função resumo é produzir um resumo a partir de um objeto qualquer como um arquivo, mensagem, etc. Conforme mostra a tabela abaixo, uma função resumo H precisa ter as seguintes propriedades [STA 99]:

- pode ser aplicada a um bloco de dados de qualquer tamanho;

[3]Parte integrante do Internet Client SDK da Microsoft®

- tem um produto de tamanho fixo como resultado;
- tem que ser relativamente fácil de calcular a função H(Ob) a partir de um dado (Ob);
- para qualquer código r calculado, tem que ser computacionalmente inviável obter (Ob) de H(Ob)=r;
- para um bloco (Ob_1) qualquer, tem que ser computacionalmente inviável obter (Ob_1) $\neq$ (Ob_2) sendo $H(Ob_1) = H(Ob_2)$;
- tem que ser computacionalmente inviável obter um par(Ob_1,Ob_2) tal que $H(Ob_1)= H(Ob_2)$.

3.6 Algoritmos de Função Resumo

Atualmente, são dois os algoritmos mais usados: MD5 e SHA-1. Ambos são derivados do popular, mas já obsoleto MD4.

3.6.1 MD5 - Message Digest Algorithm

Desenvolvido por Ron Rivest, o mesmo criador do famoso algoritmo de criptografia de chave pública, o RSA (cuja sigla significa: Rivest, Shamir e Adleman). Basicamente, este algoritmo processa blocos de 512 bits de um objeto de tamanho variável e produz um código resumo de saída de 128 bits. Este processo é feito através de cinco passos distintos:

Inserção de preenchimentos de bits: é acrescido ao objeto um preenchimento de bits de 1 até 512;

Inserção do tamanho original do objeto: é acrescido ao objeto um bloco de 64 bits que contém o seu tamanho original;

Inicialização do registro: para armazenar os resultados intermediários e finais da função resumo, é usado um registro de 128 bits, representado por 4 registradores de 32 bits;

Processamento da mensagem em blocos de 512 bits: nesta fase 4 importantes rotinas de compressão executam funções lógicas primitivas diferentes, mas com estrutura similar;

Saída: após o processamento final, o resultado é um código resumo de 128 bits.

Em termos de força como algoritmo, o MD5 possui a propriedade de cada bit no início do processo ser uma função para cada bit resultante. Em termos práticos, a dificuldade em se achar dois objetos com o mesmo resumo resultante seria na ordem de 2^{64} operações. Para obter-se um objeto a partir de um resumo gerado a dificuldade seria de 2^{128} operações. Porém, do ponto de vista da criptoanálise, o MD5 deve ser considerado vulnerável pelo ataque da força bruta ("ataque do dia do aniversário"), sendo necessário um esforço na ordem de 2^{64}. Com isto fica evidente a necessidade de se substituir o MD5 e os candidatos mais cotados são o SHA-1, SHA-256, SHA-384 e SHA-512.

3.6.2 SHA-1 - Secure Hash Algorithm

Este algoritmo foi desenvolvido pelo NIST (National Institute of Standards and Technology) e lançado como padrão em 1993 originalmente como FIPS PUB 180. Depois, em 1995, uma versão revisada do algoritmo foi lançada como FIPS PUB 180-1, tornando-o conhecido como SHA-1. Semelhante ao MD5 em termos de estrutura, o SHA-1 também em 5 passos definidos, processa blocos de 512 bits de um objeto e produz um resumo de saída de 160 bits.

Inserção de preenchimentos de bits: é acrescido ao objeto um preenchimento de bit valor 1 seguido de um número necessário de bits com valor 0, numa proporção que varia de 1 a 512 bits;

Inserção do tamanho original do objeto: é acrescido ao objeto um bloco de 64 bits que contém o seu tamanho original;

Inicialização do buffer: para armazenar os resultados intermediários e finais da função resumo, é usado um "buffer"de 160 bits, representado por 5 registradores de 32 bits;

Processamento da mensagem em blocos de 512 bits: nesta fase 4 importantes rotinas de compressão executam funções lógicas primitivas diferentes em 20 passos, mas com estrutura similar;

Saída: após o processamento final, o resultado é um resumo de 160 bits.

Se compararmos o SHA-1 com o MD5, a diferença imediata que citamos é o tamanho final do resumo com 32 bits a mais para o SHA-1. Como resultado, teríamos uma dificuldade em achar dois objetos com o mesmo resumo resultante na ordem de 2^{80} operações. Para obter-se um objeto a partir de um resumo gerado seria necessário um esforço envolvendo 2^{160} operações.

3.7 Assinatura Digital

A assinatura digital é um mecanismo de autenticação que capacita o criador de um objeto, unir a este objeto, um código que age como assinatura [STA 99]. Este código, inclui uma seqüência de caracteres gerados por funções específicas (função resumo) a partir de determinado objeto, cifrados por uma informação exclusiva de seu emissor, como por exemplo: a chave privada. Esta seqüência de caracteres pode ser unida ao objeto ou pode ser enviada separadamente. Uma assinatura digital válida, confirma que o objeto não foi alterado desde o ato de sua assinatura e identifica seu assinante. A assinatura digital não altera o conteúdo do objeto. A assinatura digital surgiu para dar autenticidade ao documento eletrônico. Ela possui propriedades que permitem [STA 99]:

1. verificar o autor, data e hora da assinatura;

2. autenticar o conteúdo do documento ou objeto na hora da assinatura;

3. fácil verificação por terceiros para resolver disputas judiciais.

Com base nestas propriedades, podemos dizer que a assinatura exige:

1. ser uma seqüência de bits que dependa do objeto a ser assinado;

2. usar alguma informação exclusiva do emissor do objeto, prevenindo que este seja negado ou forjado;

3. ser relativamente fácil produzir, verificar e reconhecer a assinatura;

4. ser computacionalmente inviável de forjar a assinatura digital, seja por construção de um novo objeto com uma assinatura já existente ou construção de uma assinatura fraudulenta para um objeto já existente;

5. ser prático de se armazenar uma cópia da assinatura digital.

Assinaturas digitais são manipuladas usando-se chave pública e chave Privada, matematicamente relacionadas e que pertencem a um único proprietário. A chave pública é disponibilizada para a comunidade, mas a chave privada permanece exclusivamente com o seu dono. O fato é que se uma das chaves é usada para a cifrar, a outra será necessária para decifrar. No caso da assinatura digital, a chave privada é usada para gerar a assinatura, enquanto a chave pública é usada para a verificação.

O Authenticode usa esta tecnologia de assinatura digital para garantir a origem e a integridade do software com assinatura digital baseada em chave pública/privada e geração do código resumo, respectivamente. A assinatura do código envolve o cálculo do resumo, o qual é cifrado e adicionado ao código. Se algo no código for mudado, seja um só bit, a integridade estará comprometida.

O código resumo gerado pelo Authenticode é de 128 ou de 160 bits, dependendo do algoritmo usado no cálculo. A figura 3.5 ilustra o processo

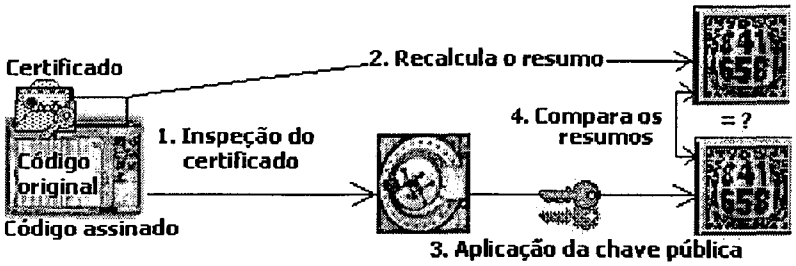

Figura 3.5: Processo de verificação para um código assinado. No processo de verificação para um código assinado, a chave pública constante no certificado digital é aplicada para decifrar o resumo que foi calculado e cifrado durante a assinatura. Além disto, um novo resumo é gerado a partir do código original. Este recálculo de um novo resumo é realizado da mesma forma que o resumo gerado no ato da assinatura. Por fim, os resumos são comparados. Se forem idênticos, a assinatura é válida.

de validação de um código assinado. Este processo é transparente para o usuário que, por exemplo, baixa um código assinado pela Internet. O próprio navegador se encarrega de validar o código, iniciando pela verificação da autenticidade da identidade digital do desenvolvedor, utilizando, para isto, a chave pública da AC que a certifica, ou seja, a identidade digital do assinante do código vem assinada pela chave privada da AC. Para decifrar o código resumo o navegador utiliza a chave pública contida na identidade digital do assinante. Após isto, o mesmo algoritmo de cálculo de resumo é usado para gerar um novo código resumo, comparando-o, assim, com o código resumo decifrado. Se os códigos forem idênticos, então a validação foi bem sucedida.

3.8 Biblioteca de Funções - CryptoAPI

A interface de programação de aplicações para criptografia da Microsoft®, a CryptoAPI [COR 01b], é um conjunto de funções exportadas por DLLs (dynamiclink libraries) residentes no sistema operacional Windows® e que permite aos desenvolvedores cifrar/decifrar dados, além de assinar objetos, trabalhar com certificação digital e manipular e proteger as chaves públicas/privadas dos usuários [COR 96a]. A advapi32.dll é responsável por cifrar e decifrar objetos e

pelo armazenamento e verificação dos certificados digitais de documentos. Por sua vez, a crypt32.dll possui funções para verificação de certificados e criação/análise de mensagens. Qualquer desenvolvedor que faça uso da CryptoAPI não precisa se preocupar muito com o manuseio de algoritmos de cifragem ou chaves públicas/privadas, pois a biblioteca se encarrega destes detalhes.

3.9 Conclusão

A assinatura de código surgiu para resgatar a confiança entre a comunidade de usuários e várias aplicações disponíveis na Internet. É a prova de fraudes pela tecnologia disponível atualmente e é uma prova incontestável de identificação do assinante. A assinatura protege os códigos e o usuário contra qualquer alteração provocada por atividade maliciosa como vírus ou alteração propositada por um indivíduo agindo de má fé.

Capítulo 4

Infra-Estrutura de Chave Pública (ICP)

4.1 Introdução

A infra-estrutura de chave pública é uma série de padrões concordantes que envolvem autoridades certificadoras, estruturas entre múltiplas ACs, métodos para achar e validar caminhos de certificação, protocolos operacionais, protocolos de gerenciamento, ferramentas inter-operantes, certificados e apoio à legislação. Em outras palavras, é um conjunto de serviços necessários quando a tecnologia baseada em chave pública é usada em grande escala. Compete aos protocolos operacionais a entrega de certificados e a lista de certificados revogados (LCR) aos sistemas (aplicações) que necessitam validar assinaturas. Os protocolos de gerenciamento são responsáveis por interações entre diferentes componentes da ICP, estando eles conectados ou não à Internet. Eles também fornecem meios para registro, inicialização, certificação, revogação e recuperação de pares de chaves. Para conceber toda esta estrutura de chave pública é necessário levar em conta alguns importantes requisitos, como:

- escalabilidade, onde um conjunto de credenciais de usuário comporta-se de forma previsível em qualquer escala de uso;

- múltiplas aplicações suportadas por infra-estruturas comuns, proporcionando conveniência, segurança e economia para os usuários finais;
- interoperabilidade de infra-estruturas administradas separadamente por se tratar de uma rede que atravessa diferentes países;
- suporte a múltiplas políticas, onde é necessário estar apto a associar diferentes políticas a diferentes caminhos de certificação;
- simples gerenciamento de riscos e limitações de responsabilidades da AC, pois é preciso saber os riscos associados ao gerenciamento de infra-estrutura de chave pública e isenção de prejuízos resultantes do uso indevido dos certificados. Todos estes fatores convergem para a necessidade de adoção de um padrão para infra-estrutura de chave pública.

Um sistema de distribuição de chaves públicas em larga escala precisa trabalhar com relacionamentos entre múltiplas ACs. A estrutura deste relacionamento varia de acordo com a comunidade usuária, natureza das aplicações que trabalham com certificados, além da área geográfica abrangente.

Neste capítulo, serão abordados temas como Autoridade Certificadora no item 4.2, Autoridade de Registro no item 4.3, Interface Pública no item 4.4, Caminho de Certificação no item 4.5, Certificado Digital no item 4.6, nomeação no X.500 no item 4.7, recomendação e formato do X.509v3 nos itens 4.8 e 4.9, com o intuito de mostrar ao leitor alguns componentes de estrutura de chave pública. Do item 4.10 ao item 4.12, veremos alguns conceitos de restrições de certificação. No item 4.13, veremos alguns padrões importantes de criptografia de chave pública. No item 4.14, concluiremos este capítulo.

4.2 Autoridade Certificadora - AC

As Autoridades Certificadoras emitem certificados digitais para entidades que precisam se identificar e garantir suas operações no mundo eletrônico.

Cada identidade digital emitida é certificada e garantida pela autoridade certificadora responsável pela emissão. Entre as principais responsabilidades de uma AC estão:

- publicar critérios de concessão de certificados;
- conceder certificados para usuários que concordem com os critérios divulgados;
- gerenciamento de certificados, o que entre outras coisas inclui: registrá-los, distribuí-los e revogá-los;
- armazenar chaves de uma maneira segura e de fácil acesso.

As obrigações legais, responsabilidades e regras de uma AC e dos proprietários de certificados, devem estar expressas no termo DPC - Declaração de Práticas de Certificação [AUS 01] e [AN 01], que pode também ser parte integrante do contrato assinado entre a AC e o proprietário do certificado. No ato do pedido, o pretendente ao certificado precisa enviar algumas informações pessoais. Estas informações podem ser enviadas através de formulário próprio e seguramente enviadas a AC, junto com a chave pública, que pode ser gerada pelo sistema local (o par de chaves). A AC deve solicitar a confirmação da veracidade das informações recebidas antes de emitir o certificado. Isto é um fator importantíssimo e a AC pode terceirizar este serviço ou até mesmo, em alguns casos, exigir que o pretendente compareça ao seu escritório, munido de documentação para comprovação. Algumas vezes, por se tratar de uma área de abrangência muito extensa, a AC pode requerer ou contar com o auxílio de uma autoridade de registro.

Se concebermos uma visão desta estrutura de forma prática, podemos estabelecer um modelo geral de uma AC. Este modelo envolveria entidades como:

- uma ou mais AC de fato que emitem e assinam o certificado digital;
- uma ou mais AR (Autoridade de Registro) que verificam as informações e enviam o pedido para a AC;

- uma ou mais Interfaces Públicas que contenham um diretório público e que tenham como principal objetivo, a comunicação com as entidades requisitantes.

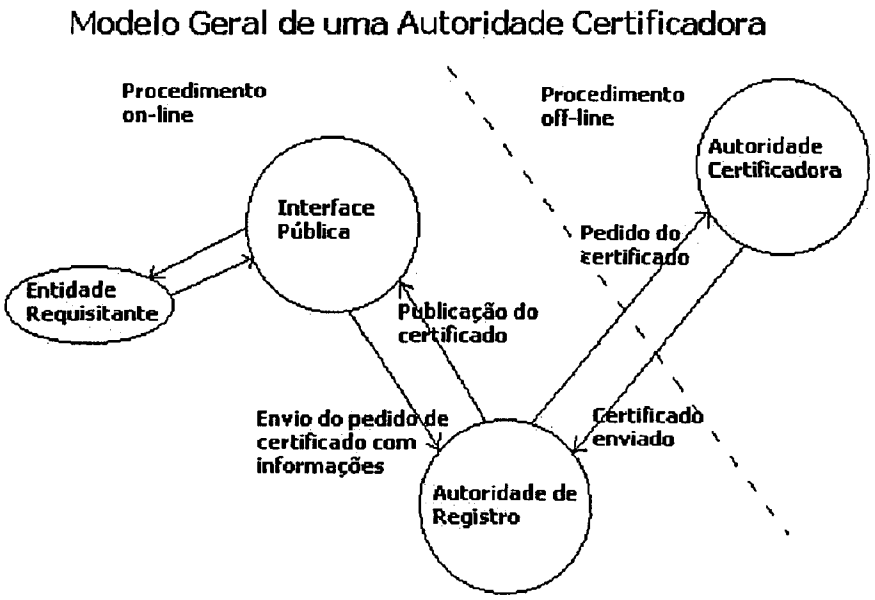

Figura 4.1: Modelo geral de uma autoridade certificadora. Esta figura ilustra um modelo geral de uma AC onde a entidade requisitante faz o seu pedido de certificado através de uma Interface Pública, num procedimento de tempo real. A Interface Pública repassa este pedido para uma AR que tem a função de verificar a veracidade das informações contidas no pedido. Só depois da aprovação da AR é que o pedido é repassado para a AC, o que pode ser feito através de uma mídia de dados, por questões de segurança. Quando o certificado fica pronto, ele faz o caminho inverso até a entidade requisitante.

Esse modelo geral pode ser melhor entendido com a ilustração da figura 4.1. É importante salientar que a comunicação entre a entidade requisitante e a Interface Pública é feita de forma interativa, onde a entidade requisitante baixa um programa para sua máquina local que emitirá seu pedido de certificação. O pedido após chegar na Interface Pública, é repassado para a AR e condicionado a uma avaliação. Só depois o pedido de certificação é enviado para a AC de uma forma segura (off-line), ou seja, não conectada e, geralmente, através de mídia de dados, tais como zipdrive, disquetes, cds, etc.

4.3 Autoridade de Registro - AR

A Autoridade de Registro é responsável pela verificação das informações fornecidas pela entidade requisitante do certificado. Ela atua como um órgão de apoio à AC e, em alguns casos, pode exigir que o requisitante compareça ao escritório da AR para garantir a veracidade das informações ou, até mesmo, terceirizar este tipo de serviço contratando empresas para irem até o requisitante. Entre as principais funções da AR, estão:

- verificar a identidade do requisitante;
- verificar autenticidade das informações;
- enviar para uma Interface Pública os certificados emitidos;
- comunicar com antecedência o fim do prazo de validade do certificado;
- verificar/receber pedidos de revogação de certificados;
- encaminhar os pedidos de certificação para as ACs;
- encaminhar para uma Interface Pública as LCRs;
- encaminhar os pedidos de revogação de certificados para as ACs.

4.4 Interface Pública

A Interface Pública exerce um papel de interação com o mundo externo ao do modelo geral de uma AC (conforme ilustra a figura 4.1). Ela provê à entidade requisitante uma forma de requerer os certificados digitais, consultar listas de revogações, etc. No caso de requisição de certificado, a entidade requisitante pode baixar um programa de emissão de certificados para sua máquina local, a qual emitirá seu pedido de certificação em conformidade com os padrões PKCS #10, criando um par de chaves (pública e privada). Entre as principais funções da Interface Pública, estão:

- publicação dos certificados, normalmente em diretório;
- manter atualizada a lista de Certificados revogados;
- prover ferramentas para requisição de certificados;
- receber os pedidos de certificação;
- manter contato com as entidades requisitantes.

4.5 Caminho de Certificação

O caminho de certificação é uma seqüência de um ou mais nós conectados entre o assinante e a sua respectiva autoridade certificadora [FEG 99], pg.85.

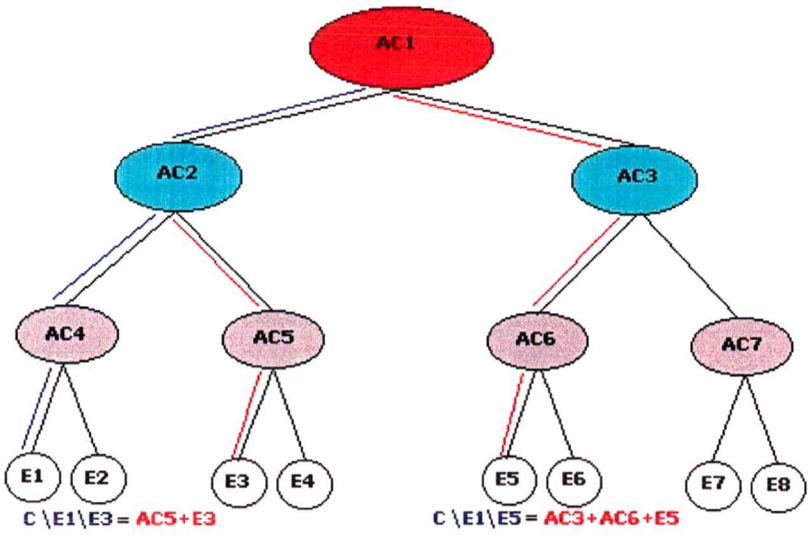

Figura 4.2: Estrutura hierárquica top-down. Digamos que a entidade E1 queira se subscrever para a entidade E3. O caminho de certificação entre E1\E3 envolveria AC5 e E3, pois para E1, AC2 é considerada como uma autoridade certificadora raiz. Da mesma forma se E1 precisasse confiar em E5, a autoridade certificadora raiz seria AC1 e o caminho de certificação E1\E5 seria composto dos nós AC3, AC6 e E5. Vale lembrar que a relação de confiança entre a entidade certificada e sua autoridade certificadora raiz é plena.

Para que seja possível percorrer um caminho de certificação e validar um certificado, as ACs formam uma corrente de ligação que pode ser chamada de estrutura entre múltiplas autoridades certificadoras. As formas de ligação mais comuns são chamadas de estruturas hierárquicas do tipo "top-down", onde no topo da hierarquia temos a autoridade certificadora raiz, seguidas de ACs subordinadas e, finalmente, as entidades certificadas. A figura 4.2 ilustra este relacionamento.

Uma variação da estrutura ilustrada pela figura 4.2, seria uma hierarquia com ilhas de confiança entre ACs do nível mais alto (veja a figura 4.3). Este tipo de hierarquia, chamado de hierarquia de floresta de certificação, funcionam melhor em comunidades que não possuam um padrão bem formado de estrutura, como a comunidade da Internet. Esse tipo de estrutura, com interconexões entre diferentes ACs, suportam uma grande demanda de aplicações de chave pública, como, por exemplo, e-mail seguro.

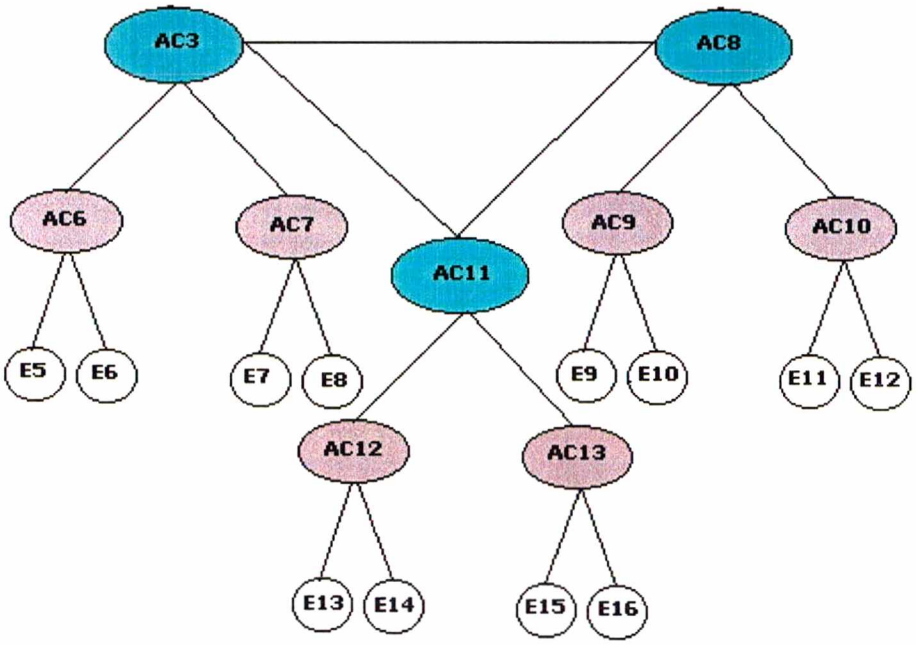

Figura 4.3: Estrutura Hierárquica de Floresta de Certificação. Neste tipo de estrutura, cada comunidade, baseada em sua localização geográfica, confia em uma particular AC. As ACs no topo de cada floresta, estão ligadas entre si, formando uma corrente de ligação entre diferentes posições geográficas.

4.6 Certificados Digitais

O certificado digital ou identidade digital é uma forma de credenciamento eletrônico. Ele é emitido por uma entidade, chamada de AC, e que estabelece uma identidade para o solicitante. A tecnologia utilizada na identidade digital é totalmente baseada na tecnologia de par de chaves pública/privada, onde a chave pública é armazenada na identidade digital. No caso, qualquer código pode ser assinado utilizando-se a chave privada do seu desenvolvedor e somente poderá ser validado com a chave pública correspondente ao mesmo par. O propósito do certificado digital é manter uma relação de confiança entre a chave pública e o proprietário dela. Quando uma AC emite um certificado digital, ela garante, sob risco de prejuízo de sua reputação, que o proprietário do certificado é realmente quem diz ser.

A entidade certificada pode ser uma pessoa, um dispositivo de hardware, como, por exemplo, um roteador ou mesmo um software. Diferente dos órgãos competentes emissores de documentos de identidade, passaportes, licenças de motorista, etc., que precisam utilizar papel em alto relevo, brasões e outros recursos para dificultar a falsificação e aumentando assim o custo desses documentos, o certificado digital pode ser emitido por software relativamente simples de ser feito. A ferramenta de programação "keytool", usada na linguagem de programação JAVA™ e a CryptoAPI do Windows®, são exemplos típicos de componentes que podem ser usados na criação de software emissor de identidade digital.

A AC depois de emitir o certificado para uma entidade, ela assina "digitalmente" o certificado, incorporando a assinatura da AC.

4.7 Nomeação no X.500

Como a recomendação X.509 foi originalmente desenvolvida para funcionar no sistema de diretórios do X.500 [UNI 97a] e [TAH 99], é necessário um pouco de conhecimento sobre este sistema para que tenhamos um entendimento

melhor do X.509.

O X.500 trata os diretórios de uma maneira muito parecida com um catálogo telefônico. A partir do nome de uma pessoa é possível se achar outras informações relativas a esta pessoa. Claro que no sistema de diretórios do X.500 é possível obter-se muito mais do que endereço e número de telefone. Uma entrada no diretório X.500 pode representar qualquer entidade real, não só pessoas, mas computadores, companhias, governos, nações, etc..., além de poder abrigar os certificados especificando as chaves públicas das entidades.

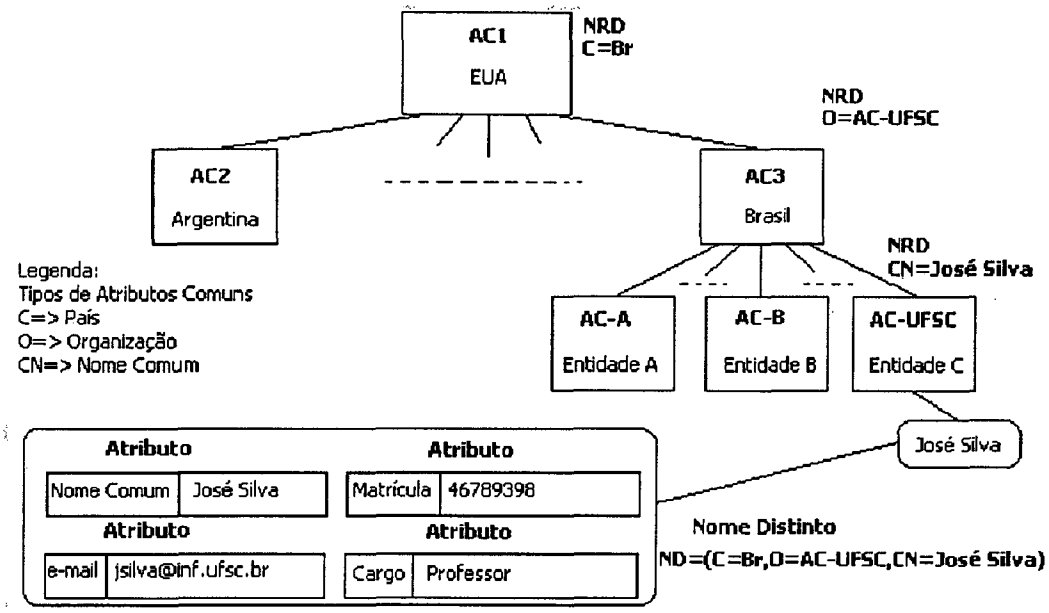

Figura 4.4: Exemplo de construção de nomes no X.500. Neste exemplo, digamos que a AC1 seja a autoridade certificadora máxima para todos os países e que a AC3 seja a autoridade certificadora máxima no país Brasil. Digamos que a AC3 certifica a AC-UFSC e que este emita certificados para o domínio inf.ufsc.br. Para chegarmos ao ND completo de José Silva teríamos que concatenar os NRDs seguindo a hierarquia da árvore, a começar pela raiz (que por default tem um ND nulo). Então teríamos: ND de José Silva (C=Br,O=AC-UFSC,CN=José Silva). Existe um problema com o atributo CN, pois José Silva é um nome muito comum e passível de ambigüidade. Além disto, José Silva pode sair da entidade UFSC e tempos mais tarde um outro José Silva poderia inadvertidamente herdar os privilégios de José Silva original. Uma boa política para se garantir um NRD único é adotar um atributo que não seja reutilizável ou ambíguo. Neste caso, poderíamos agregar o atributo "matrícula" (EN-Employee Number) com o nome comum e aí teríamos um DN garantido (C=Br,O=AC-UFSC,CN=José Silva,EN=46789398).

É importante saber que cada uma dessas entradas está associada a um nome único que é tecnicamente chamado de nome distinto - ND, conforme ilustra a figura 4.4. O diretório é organizado de uma maneira hierárquica chamada de ADI (Árvore de Diretório de Informação), onde somente o raiz, elemento mais acima na árvore, não possui um pai (elemento imediatamente superior). Cada elemento, com exceção do raiz, está associado a um nome distinto relativo chamado NRD (Nome Relativo Distinto). O NRD se torna cada vez maior a medida que se aproxima da base da árvore, pois é resultado da concatenação dos nomes imediatamente superiores.

4.8 Recomendação X.509

A proposta para implementação de uma infra-estrutura de chave pública (ICP) apareceu em 1988. É a mais antiga proposta que se tem conhecimento para implementação de ICP e foi concebida para trabalhar nas recomendações do serviço de diretórios X.500 (parte da série ISO/ITU que trata de serviços de diretório para redes de computadores de grande abrangência).

4.9 Formato do Certificado X.509 (versão 3)

O X.509 evoluiu para a versão 3 a partir de junho de 1997, com a conclusão do relatório final da recomendação da ITU-T [UNI 97b] e [TAH 99]. Com as versões 1 e 2, constatou-se que os formatos dos certificados eram deficientes em vários aspectos e precisavam carregar informações adicionais para o padrão tornar-se mais seguro e eficaz. A fundamental mudança foi fazer o formato do certificado e do LCR tornar-se extensível para poder carregar informações como política fornecida, atributos do proprietário e emissor do certificado, restrições de caminho de certificação, etc. Como exemplo disto, podemos citar a necessidade de se incluir nas extensões uma lista de políticas seguidas para a criação do certificado, de modo a assegurar que determinado certificado criado para troca de mensagens de e-mail não

seja usado em transações financeiras (a figura 4.5 ilustra o formato do certificado X.509 e suas extensões) [FOR 97].

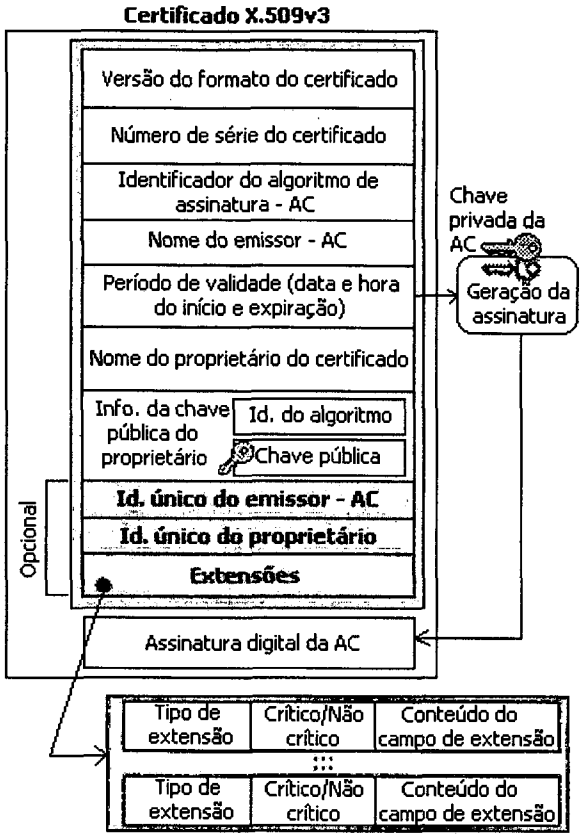

Figura 4.5: Estrutura do certificado X.509 versão 3. Na figura acima, podemos observar o formato de um certificado x.509v3 com os seguintes campos: **1.Versão:** contém o número da versão do certificado, que atualmente pode ser 1, 2 ou 3; **2.Número de série:** Número único emitido pela AC que identifica o certificado; **3.Identificador do algoritmo de assinatura:** campo que identifica o algoritmo usado pela AC para assinar digitalmente o certificado; **4.Nome da AC emissora:** identifica a AC que emitiu e assinou o certificado; **5.Período de validade:** contém o período em que o certificado é válido; **6.Nome do proprietário:** nome do proprietário dono da chave pública contida no certificado. Este nome é único, porque para cada certificado pode haver apenas um dono; **7.Informação da chave pública do proprietário:** campo que carrega a chave pública do proprietário e o identificador do algoritmo usado por ela; **8.Identificador único do emissor:** identificador utilizado para evitar possíveis ambigüidades com o nome da AC emissora; **9.Identificador único do proprietário:** identificador utilizado para evitar possíveis ambigüidades com o nome do proprietário; **10.Extensões:** campos que provêem meios de associar informações adicionais para uso de aplicações de ICP.

No caso das extensões voltadas para restrição de caminho de certificação, isto possibilita uma autoridade certificadora (AC) impor condições para evitar possíveis fraudes, como entidades fim se passarem por ACs. Uma AC pode também impor uma restrição progressiva de confiança para prevenir a formação de caminhos de certificação infinitos.

4.10 Restrições Progressivas de Confiança

O modelo de restrições progressivas de confiança permite que qualquer autoridade certificadora estabeleça limites ou condições para o propósito do certificado. Como resultado, os usuários terão menos chances de se deparar com caminhos de certificação muito longos, diminuindo os riscos de erros, falsificações e certificações não apropriadas. Este conceito é mostrado pela figura 4.6. O usuário Paul tem como sua autoridade certificadora a AC1 , entidade pela qual ele (Paul) confia plenamente.

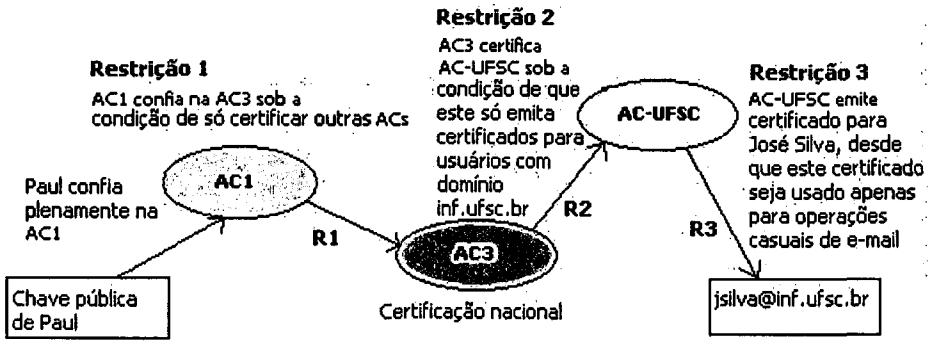

Figura 4.6: Corrente de uma Restrição Progressiva de Confiança. No exemplo acima, Paul, que está na extrema esquerda da corrente, sabe que o certificado de José Silva, na extrema direita, serve apenas para operações de correio eletrônico.

Por sua vez, a AC1 certifica outra AC chamada AC3, delegando que a AC3 somente possa emitir certificados para outras ACs. Seguindo a corrente, a AC3 certifica a AC-UFSC para que este possa certificar usuários finais pertencentes ao domínio inf.ufsc.br. Agora, a AC-UFSC estabelece que o certificado emitido para

o usuário José Silva é para fins de correspondência eletrônica, excluindo seu uso em operações comerciais. Conforme visto, a confiança que era máxima em um extremo da corrente tornou-se restrita no outro extremo. Desta maneira, Paul sabe que o certificado de José Silva é para uso somente de correspondência eletrônica. Em tempo, Paul também não aceitaria nenhum certificado emitido pela AC3 para José Silva ou outro usuário final, bem como não aceitaria nenhum certificado de qualquer autoridade certificadora certificada pela AC-UFSC.

4.11 Restrição de Caminho de Certificação

4.11.1 Requerimentos

De uma maneira resumida e de acordo com a recomendação do padrão X.509 versão 3 da ITU-T [UNI 97b], pg.34, para o processamento do caminho de certificação é necessário:

- que o certificado emitido para ACs sejam diferenciados de certificados emitidos para entidades fim, de modo a inibir que entidades fim se façam passar por ACs. Que seja possível para a AC limitar o tamanho da corrente subseqüente resultante de um certificado de uma AC, para, por exemplo, não mais que um ou dois certificados;

- uma AC precisa ser capaz de especificar restrições que permitam ao usuário do certificado verificar se a AC menos confiável em um caminho de certificação não está emitindo certificados fora do domínio imposto;

- o processamento de caminho de certificação precisa ser implementado em um módulo independente e automático para permitir que a implementação em hardware e software sejam confiáveis;

- que o usuário não precise depender de interação em tempo real para estas implementações;

- que seja possível implementar o processamento de caminho de certificação sem depender de bases de dados locais de política e descrição de informações;
- o caminho de certificação precisa operar em ambientes dos quais múltiplas políticas de certificados sejam reconhecidas;
- é requerido completa flexibilidade no modelo de confiança, tanto para simples organizações como para múltiplas empresas interconectadas e com o caminho de certificação começando em um domínio local seguro;
- estruturas de nomes não devem ser forçadas pela necessidade de uso dos nomes nos certificados. Ex.: estrutura de nome de diretórios x requerimentos de ACs;
- campos de extensões de certificados devem manter compatibilidade com versões anteriores da ITU-T Recomendação X.509—ISO/IEC 9594-8;
- uma AC precisa ser capaz de inibir o uso de mapeamento de política e de requerer explicitamente identificadores de política de certificados.

4.12 Campos de Extensão de Certificados

4.12.1 Restrições Básicas

Este campo indica se a entidade do certificado é uma AC ou tão somente um usuário final. Ele é importante para detectar possíveis fraudes de entidades se passando por ACs. Se realmente a entidade for uma AC, a restrição de tamanho de caminho de certificação também deve ser indicada.

Sintaxe [1] :

BasicConstraints EXTENSION ::={

SYNTAX BasicConstraintsSyntax

[1] Maiores informações a respeito da sintaxe e da linguagem empregadas nos exemplos a seguir, podem ser obtidas em [UNI 97b]

IDENTIFIED BY id-ce-basicConstraints}

BasicConstraintsSyntax ::= SEQUENCE {
cA BOOLEAN DEFAULT FALSE,
pathLenConstraint INTEGER (0..MAX) OPTIONAL}

Exemplo: Digamos que a AC-UFSC e AC-A, ambas autoridades certificadoras, quisessem emitir certificados uma para outra. Porém, a AC-UFSC deseja que a comunidade da UFSC somente aceite certificados de usuários finais emitidos por AC-A e não por outras ACs certificadas por AC-A. Neste caso, AC-A teria que emitir certificados com a seguinte extensão:
{ cA TRUE, pathLenConstraint 0 }

A variável pathLenConstraint com valor zero indica que a corrente não deve se estender para além de AC emissora.

4.12.2 Restrições de Nomes

Este campo restringe a área dentro da qual todos os nomes das entidades dos certificados para um caminho de certificação devem estar localizados.

Sintaxe:

NameConstraints EXTENSION ::= {
SYNTAX NameConstraintsSyntax
IDENTIFIED BY id-ce-nameConstraints}

NameConstraintsSyntax ::= SEQUENCE {
permittedSubtrees [0] GeneralSubtrees OPTIONAL,
excludedSubtrees [1] GeneralSubtrees OPTIONAL}

GeneralSubtrees ::= SEQUENCE SIZE (1..MAX) OF GeneralSubtree

GeneralSubtree ::= SEQUENCE {
base GeneralName,
minimum [0] BaseDistance DEFAULT 0,
maximum [1] BaseDistance OPTIONAL}

BaseDistance ::= INTEGER (0..MAX)

Exemplo: Vamos supor que a AC-UFSC e a AC-A de Manaus, Brasília e Bahia queiram emitir certificados uma para outra, mas que a AC-UFSC imponha condições seguindo os seguintes critérios:

- para a AC-A de Manaus todas as operações são aceitas exceto operações de compra;
- para a AC-A da Bahia somente operações que estão subordinadas à matriz em Manaus são aceitas;
- para a AC-A de Brasília todas as operações são aceitas exceto àquelas subordinadas diretamente a empresa XYZ.

Então teríamos:
{cA TRUE}
{permitedSubtrees {{base -C=Br, O=CA-A de Manaus-},
{base -C=Br, O=CA-A da Bahia-, maximum 1},
{base -C=Br, O=CA-A de Brasília-}},
excludedSubtrees {{base -C=Br, O=CA-A de Manaus, OU=Compra-},
{base -C=Br, O=CA-A de Brasília, OU=XYZ-, minimum 2}}}

4.12.3 Restrições de Política

São restrições que pedem explicitamente identificação da política do certificado ou pedem inibição da política de mapeamento do resto do caminho

de certificação.

Sintaxe:

PolicyConstraints EXTENSION ::= {
SYNTAX PolicyConstraintsSyntax
IDENTIFIED BY id-ce-policyConstraints}

PolicyConstraintsSyntax ::= SEQUENCE {
requireExplicitPolicy [0] SkipCerts OPTIONAL,
inhibitPolcyMapping [1] SkipCerts OPTIONAL}
SkipCerts ::= INTEGER (0..MAX)

Exemplo: Supondo que a AC-UFSC representando a UFSC, firme um acordo de certificação com a Universidade de Santiago (UFS) no Chile, com as seguintes condições:

- a UFS quer certificar assinaturas da AC-UFSC que dizem respeito a política chamada UFS/AC-UFSC -Comércio;
- a AC-UFSC tem uma política equivalente chamada AC-UFSC/UFS-Comércio, a ser considerada pela UFS;
- a UFS quer garantias de que todos os certificados da AC-UFSC apresentem, explicitamente, mecanismos de aplicação desta política e que inibam mapeamento para outras políticas dentro do domínio da UFSC.

A UFS poderia emitir um certificado para a AC-UFSC, com as seguintes extensões:

Valor do campo de política de certificado:
{{ policyIdentifier -OI de UFS/AC-UFSC -Comércio–}}
Valor do campo de política de mapeamento:
{{ issuerDomainPolicy - OI de UFS/AC-UFSC -Comércio–,

subjectDomainPolicy - OI de AC-UFSC/UFS-Comércio–,
Valor do campo de política de restrição:
{{ policySet { –OI de UFS/AC-UFSC -Comércio–}, requireExplicitpolicy (0),
inhibitPolicyMapping (0)}}

4.13 PKCS - Padrão de Criptografia de Chave Pública

A primeira publicação do PKCS saiu em 1991 e, atualmente, este Padrão [2] encontra-se amplamente difundido. Ele foi criado pelos laboratórios da RSA e tem como objetivo promover o desenvolvimento de aplicações seguras baseados na criptografia de chave pública.

4.13.1 PKCS #7 v1.6 - Padrão de Sintaxe de Criptografia de Mensagem

O Padrão de Sintaxe de Criptografia de Mensagem [KAL 98b] surgiu para definir diversas maneiras de se cifrar uma mensagem, estando ela com assinatura digital ou não. O uso deste padrão não ficou limitado só para mensagens eletrônicas, mas está sendo usado também em transações eletrônicas como: pagamentos com cartões de bancos, iniciativa de assinatura digital W3C [3] e outro padrão, o PKCS #12 - Padrão de Sintaxe de Intercâmbio de Informação. Atualmente, a versão 1.6 [KAL 97] do PKCS #7 está sendo trabalhada para atender as especificações do SET e, com isto, deve encerrar a série 1.x do padrão. O laboratório da RSA já pensa em começar a trabalhar na versão 2.0.

[2] O termo padrão foi utilizado como tradução da palavra em inglês "standard", porém, não significa que seja um padrão no Brasil.

[3] World Wide Web Consortium. W3C Digital Signature Initiative. Disponível em http://www.w3.org/pub/WWW/Security/Dsig.

O padrão possui 9 seções que especificam os tipos, sintaxe geral, seis tipos de conteúdo, que são: dados (data), dados assinados (signed data), dados envelopados (enveloped data), dados assinados e envelopados (signed-and-enveloped data), dados resumidos (digested data) e dados cifrados (encrypted data), e ainda especifica os identificadores de objetos.

4.13.2 PKCS #10 v1.7 - Padrão de Sintaxe de Requisição de Certificação

Inicialmente, concebido para dar suporte a criptografia de mensagens do padrão PKCS #7, o PKCS #10 descreve a sintaxe de pedidos de certificação [EAS 00] e [KAL 98a]. Um pedido de certificação consiste de um nome distinto, uma chave pública e um conjunto de atributos opcionais. Este pedido, que deve estar devidamente assinado pelo requisitante, é enviado para uma autoridade certificadora que irá transformá-lo em um certificado no padrão X.509. Existem duas razões para o envio do conjunto de atributos:

- fornecer outras informações sobre a entidade requisitante ou uma "contra-senha", na qual a entidade pode depois requisitar a revogação do certificado;
- fornecer atributos para inclusão no certificado X.509.

A requisição de um certificado consiste de três partes:

- informação da requisição de certificação, que pode conter o nome distinto da entidade, a chave pública da entidade e um conjunto de atributos adicionais;
- um identificador do algoritmo da assinatura;
- a assinatura digital da informação da requisição de certificação.

4.14 Conclusão

A infra-estrutura de chave pública é algo complexo e que envolve vários componentes, recomendações e padrões. A recomendação X.509v3, por si só, já é bastante complexa. Porém, toda esta estrutura criada para atender as necessidades de uma comunidade global é de extrema importância para dar suporte à certificação digital e manutenção destes certificados.

Capítulo 5

Modelo de Proteção de Software por Certificação Digital

5.1 Introdução

Neste capítulo, apresentaremos o modelo proposto e os resultados obtidos com o estudo de campo e a implementação do protótipo. Para começar, no item 5.2 definiremos alguns componentes necessários para a viabilização do Modelo. Em seguida, no itens 5.3 e 5.4, entenderemos como é feito o processo de licenciamento de um software protegido por certificação digital, bem como a funcionalidade do Modelo . No item 5.5 veremos em detalhes como ocorre o processo de validação, o qual foi dividido em três partes. Veremos, também, no item 5.6 que quando a validação é bem sucedida ocorre a personalização do software, com base nas informações constantes no certificado de licença. Já no item 5.7 abordaremos algumas vulnerabilidades do Modelo.

A parte prática da pesquisa, começa a partir do item 5.8, onde apresentaremos um protótipo do Modelo, com destaques para a sua construção e implementação. Por último, no item 5.9, concluiremos este capítulo.

5.2 Componentes para Viabilização do Modelo

Para que o modelo concebido tenha funcionalidade, alguns componentes tiveram que ser providenciados:

1. adoção de um padrão ASN-1 pré-definido e registrado pelo LabSEC [1], sob número: 1.3.6.1.4.1.7687.1.8. Este número representa a OID [2] destinada ao Modelo de Proteção de Software por certificação Digital e pode ser melhor entendida através da ilustração da figura 5.1;

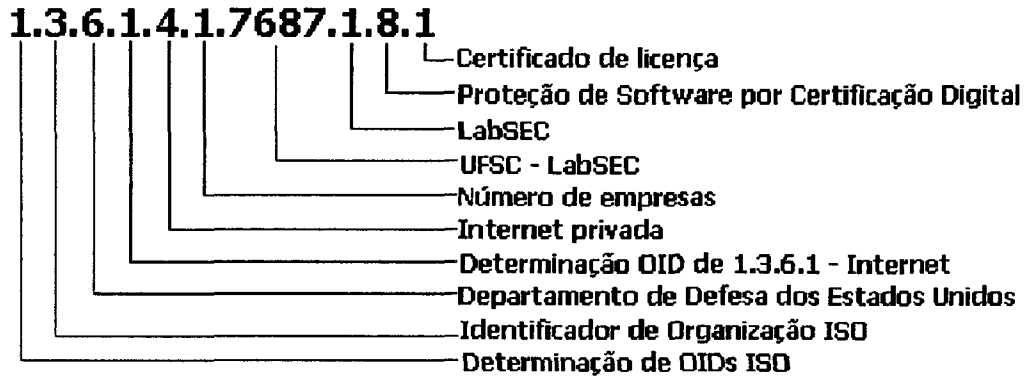

Figura 5.1: Estrutura da OID. Conforme ilustra a figura acima, cada prefixo da OID tem seu significado estabelecido.

2. definição das extensões que conterão as restrições do uso destes certificados, como por exemplo, certificados destinados a licença de uso de determinado software de alguma empresa;

3. emissão de certificados digitais para teste do protótipo. Estes certificados são baseados na recomendação X.509v3 e contém as extensões necessárias para o uso do software.

[1] LabSEC - Laboratório de Segurança em Computação - www.labsec.ufsc.br

[2] OID - Object Identifier. Para maiores informações, veja RFC 1778 e 2252.

5.3 Processo de Licenciamento do Software

Na prática, a requisição dos certificados ficaria a encargo do produtor do software ou da revenda que repassariam os dados para uma AC conveniada. O produtor ou a revenda funcionariam então como uma AR (Autoridade de Registro), com a responsabilidade de verificar a veracidade dos dados informados pelo cliente. Este ciclo está ilustrado na figura 5.2.

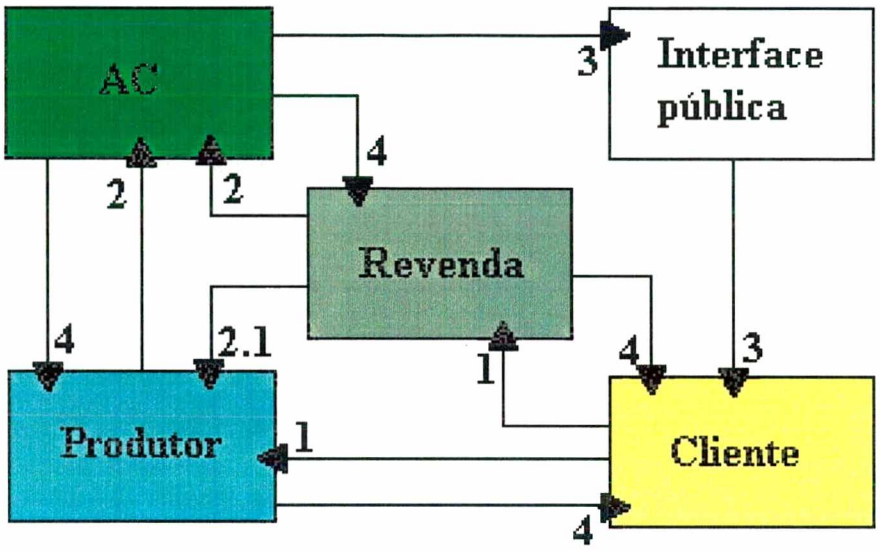

Figura 5.2: Requisição de certificados pelo produtor/revenda. O cliente, após aprovar o programa, solicita, através do produtor do software ou de uma revenda, o certificado de licença. Uma vez que este certificado seja instalado na máquina do cliente, ele estará apto a usar o programa com todas as funções habilitadas.

Numa situação hipotética, podemos imaginar que um cliente em potencial baixe, direto da página do produtor, determinado software e que decida adquirir a licença de uso. Podemos imaginar, também, que o cliente já conheça o software e o tenha comprado direto de uma revenda. Para ilustrar esta situação, poderíamos seguir o fluxo apresentado na figura 5.2:

- **setas 1:** o cliente solicita através de formulário específico, o certificado digital de licença, comprovando os dados informados. Isto poderia ser feito tanto na

revenda onde adquiriu o produto, como na página do produtor onde tenha efetuado o download do software. Em ambos os casos, tanto revenda como produtor funcionariam como uma AR;

- **setas 2:** produtor e revenda se encarregam de verificar os dados fornecidos na requisição e os submetem a uma AC de sua confiança;

- **seta 2.1:** para o caso de uma requisição através de uma revenda, esta pode também informar ao produtor a venda efetuada para que este atualize seu banco de dados com informações de mais um cliente registrado;

- **setas 3:** o certificado é emitido e, então, disponibilizado pela AC em um diretório público para que o cliente possa obtê-lo;

- **setas 4:** o certificado pode ser entregue pela revenda ou pelo produtor, caso isto signifique maior comodidade para o cliente.

Para viabilizar todo este processo de licenciamento de software, o produtor, naturalmente, precisaria criar recursos dentro de seu modelo administrativo, como por exemplo: criação de novos departamentos, contratação de pessoal, treinamento e busca de parcerias com revendas e Autoridades Certificadoras. As revendas e ACs também necessitariam adaptar-se a este novo modelo. Claro que cada empresa poderia elaborar e implementar seu próprio processo de licenciamento de software, mas, basicamente, ele giraria em torno do que foi ilustrado na figura 5.2.

Modelos como "Um Modelo Genérico de Pagamento Eletrônico com Suporte a Múltiplas Transações Comerciais" [YH 99] (veja 2.3.1), poderiam ser combinados com esta forma de proteção de software, fortalecendo o sucesso comercial dos produtos vendidos e minimizando o trabalho administrativo criado nas empresas.

5.4 Modelo Teórico de Proteção de Software por Certificação Digital

Como vimos, o usuário, após testar o software, pode adquirir o certificado digital para habilitar todas as funções do software em questão. Quando receber ou baixar de uma interface pública o seu certificado, o usuário poderá importá-lo para o sistema operacional, ficando, desta forma, apto a validar o software protegido.

Neste modelo, o software protegido faz chamadas para o sistema operacional pedindo pela validação do certificado. A gerência de certificados, um módulo de funções residentes do sistema operacional, é então acionada e **se encarrega da validação**. Este procedimento [3] poderia ser realizado através de uma consulta a um diretório público para verificar se não há revogações do certificado, conforme ilustrado pelo ponto 3c da figura 5.3, ou caso se trate de um computador não conectado à Internet, a simples validação da data de expiração do certificado. Na verdade, em qualquer uma das hipóteses, se a validação falhar, o software não é acionado. O fato de ser o sistema operacional, através da gerência de certificados, e não o software protegido, o encarregado de validar as operações ligadas ao certificado, dificulta a ação de usuários sofisticados que tentam "quebrar" o código para retirar a proteção.

As vantagens que a Proteção de Software por Certificação Digital podem oferecer ao produtor de software são grandes, pois, além de proporcionar forte proteção de cópia baseada na tecnologia de certificação e assinatura digital, ela fornece um maior controle sobre as cópias ilegais. Com a ajuda da figura 5.3,

[3] Além desse procedimento debatido aqui, outras formas de validação executadas pela gerência de certificados estão implícitas. Também é feita, por exemplo, a validação de: i) cada certificado constante na corrente de certificação do certificado de licença; ii) restrições progressivas de confiança e de caminho de certificação; iii) LCRs para todos os certificados envolvidos na operação. Informações mais completas sobre validação de certificados, podem ser encontradas em: [UNI 97b], [FEG 99] e [AN 01].

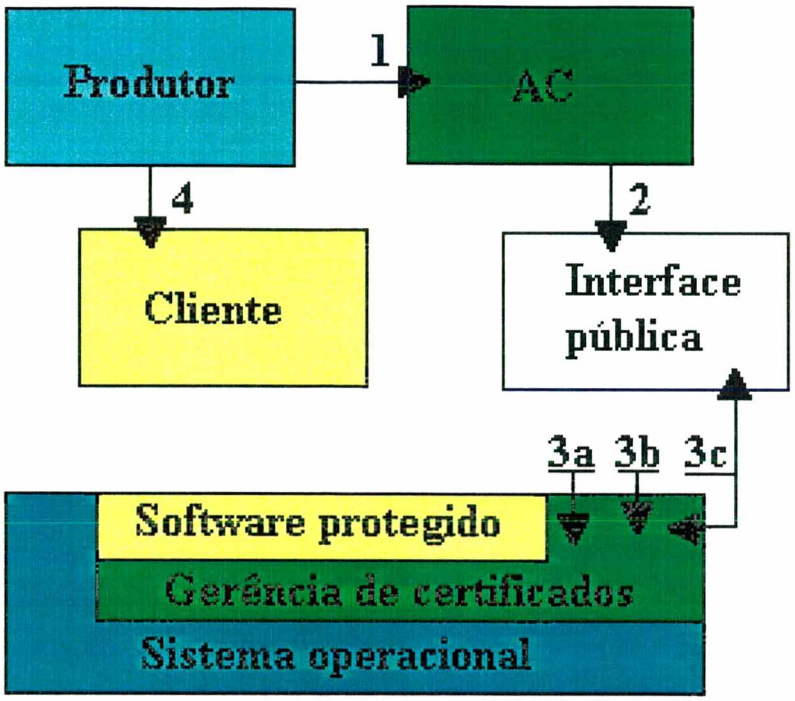

Figura 5.3: Esquema de proteção de software por certificação. O Modelo de Proteção de Cópia por Certificação Digital oferece ao produtor de software, além da proteção de cópia baseada na tecnologia de certificação digital, um maior controle sobre as cópias ilegais. Quando o produtor descobre cópias ilegais do software no mercado, ele pode: 1: solicitar a AC que revogue os certificados que fazem funcionar as cópias ilegais; 2: a AC então publica a lista de certificados revogados (LCR) com os respectivos certificados; 3 a,b,c: quando a cópia ilegal for acionada a validação ocorre através da gerência de certificados, e esta, verifica se o certificado continua na data de validade ou se não foi revogado; 4: antes de proceder a revogação, o produtor pode dar nova chance aos clientes infratores, fornecendo-lhes novos certificados de licença.

podemos imaginar a seguinte situação:

Após um certo número de licenças vendidas, o produtor de software decide fazer uma pesquisa para saber se o número de licenças vendidas se aproxima do número de licenças instaladas no mercado. Esta pesquisa poderia ser feita através de auditorias contratadas ou em conjunto com órgãos de combate à pirataria, tais como: ABES, órgãos públicos, etc. Digamos que a pesquisa revele que o número de licenças instaladas é maior que o número de licenças vendidas e que há um número

muito grande de licenças especificamente de dois clientes espalhadas pelo mercado. Segundo Austin, *"Embora não diretamente ligados a funções e responsabilidades de uma AC, outras partes envolvidas num conjunto de ICP e diretamente ligadas à AC, possuem sua parcela de responsabilidade"* [AUS 01]. O proprietário do certificado, o qual possui implicitamente sua identidade e chave pública ligadas ao certificado, possui sua parcela de responsabilidade dentro de uma ICP e deve comunicar imediatamente à AR em que se encontra credenciado, qualquer suspeita que comprometa sua chave privada - *"Qualquer incidência ou suspeita que comprometa a chave privada, resulta numa imediata notificação à AR"* [AUS 01]. Este procedimento pode ser naturalmente comparável ao de um cliente de cartão de crédito, onde a perda ou roubo deste cartão deve ser comunicada à operadora do cartão imediatamente. O produtor pode, então, suspeitar que estes dois clientes quebraram com o termo de contrato do software e desencadearam uma série de cópias ilegais no mercado, distribuindo junto com essas cópias, os certificados e respectivas chaves privadas. Diante desta situação, o produtor pode, além de tomar medidas legais cabíveis (baseado no não repúdio da assinatura digital [STA 99]), proceder da seguinte maneira (conforme ilustração da figura 5.3):

- **seta 1:** o produtor solicita à AC que revogue os certificados pertencentes aos 2 clientes em questão;
- **seta 2:** a AC publica na lista de certificados revogados (LCR) os respectivos certificados;
- **seta 3a:** toda vez que o software protegido for acionado, a validação irá ocorrer. Esta será feita através da gerência de certificados, que se encarrega de verificar se o certificado expirou ou;
- **seta 3b:** se o certificado está incluído na LCR local do sistema operacional ou;
- **seta 3c:** se o certificado está incluído na LCR publicada pela AC. Para entender melhor este processo exposto pelos pontos 3a, 3b e 3c, veja o item 5.5;

- **seta 4:** antes de efetuar realmente a revogação, seria prudente que o produtor comunicasse aos usuários das cópias legais sobre esta revogação, suas razões e que fornecesse prazos para prova de inocência e também certificados "provisórios" para uso destas cópias legais dentro deste prazo. Uma vez comprovada a não participação do cliente no comprometimento da chave privada, este receberia um novo certificado de licença.

Vale observar que no momento em que o certificado é incluído na LCR, todos os computadores com cópias ilegais e que estejam ligados à Internet deixarão de funcionar, uma vez que será feita uma consulta a esta lista. Claro que esta função pode ser implementada de maneira que o usuário fique ciente destas consultas e que ele possa ter o controle de autorizá-las ou não. Os demais computadores com cópias ilegais e que por ventura não tenham conexão com a Internet, terão seus certificados espirados naturalmente pela data de validade do certificado.

Além da descoberta de cópias ilegais através de auditorias, é viável a construção de um *módulo de denúncias* a ser implementado junto com o Modelo. Esta seria mais uma grande vantagem da Proteção de Software por Certificação Digital, pois qualquer usuário que suspeite estar usando uma cópia pirata poderia efetuar denúncia através de correspondência eletrônica. Esta idéia, abordaremos como trabalhos futuros no capítulo 6.

5.5 Processo de Validação da Licença de Uso do Software

Neste item veremos com maiores detalhes como o software protegido irá validar a licença de uso.

Para um melhor entendimento, o Processo de Validação da Licença de Uso do Software será apresentado em três partes distintas. A primeira parte tratará desde o processo de autenticação do sistema operacional até a assinatura

do desafio [4], a segunda parte cuidará da validação da assinatura feita e a terceira e última parte será responsável pelo controle de validade do certificado e pela consulta às LCRs (local e remota).

5.5.1 Parte 1

Este estágio começa quando o usuário efetua a autenticação ("logon") de sua seção no sistema operacional, preparando a liberação da chave privada do usuário que se encontra cifrada em um repositório ou em um dispositivo de hardware. Depois, quando o usuário aciona o software protegido, este interage de forma segura com a gerência de certificados, para verificar se existe na coleção [5] de certificados do usuário, a presença de algum certificado com extensão que indique seu uso para a execução de software protegido, além de verificar se este certificado contém uma chave privada correspondente.

Com o certificado de licença e a chave privada disponíveis, o software realiza, ainda através da gerência de certificados, o que seria a criação de um desafio para o usuário. Este desafio é gerado e assinado com a chave privada. Se no momento da instalação deste certificado pelo sistema operacional, o usuário optar por informar uma senha de autorização de uso da chave privada, então no ato da assinatura digital, será necessário informar esta senha para que a chave privada possa ser usada na assinatura do desafio.

A figura 5.4 ilustra bem os eventos descritos nesta etapa. Note que a requisição da senha para uso da chave privada é o único evento que requer uma interação com o usuário. O restante do processo é passado desapercebido.

---

[4] Chama-se de desafio um número identificador a ser usado uma única vez para uma situação particular [STA 99].

[5] Um usuário pode ter mais de um certificado, para várias finalidades, geralmente armazenados na pasta "my store" do perfil do usuário no sistema operacional, formando, assim, uma coleção de certificados.

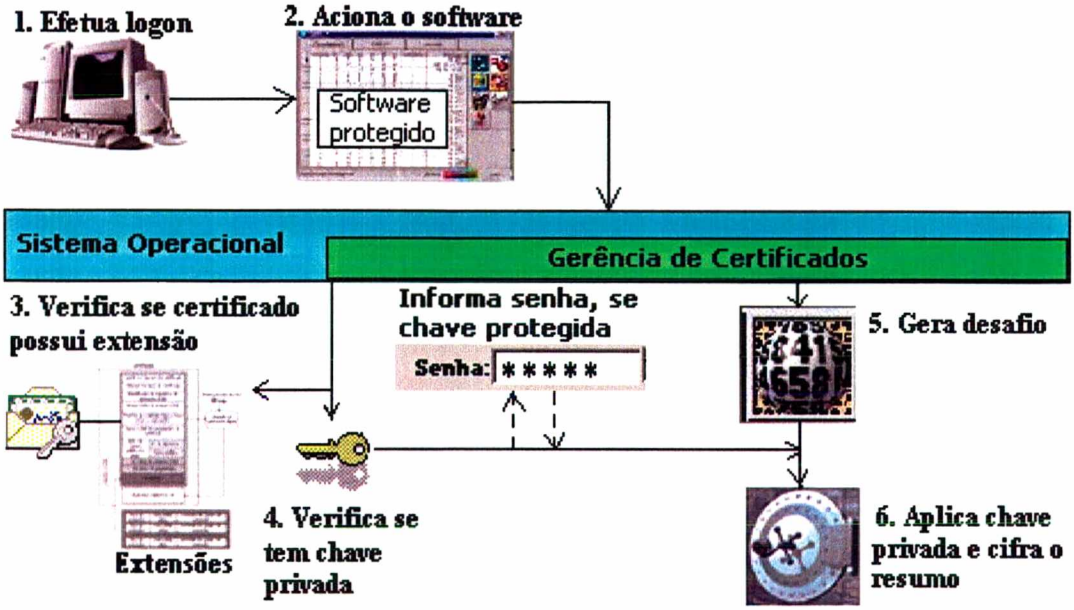

Figura 5.4: Processo de validação da licença de uso do software - parte 1. O processo de validação começa com a liberação da chave privada, a partir da autenticação do usuário no sistema operacional (1). O usuário aciona (2) o software protegido e este solicita que a gerência de certificados verifique (3) se existe algum certificado que possua uma extensão própria para o uso do software, e ainda, se este certificado possui uma chave privada correspondente (4). Em caso positivo, um desafio é gerado (5) e depois assinado (6) com a chave privada, encerrando a primeira parte do processo de validação. Como a chave privada é usada nesta primeira parte, existe a possibilidade de o usuário ter que informar uma senha de autorização de uso da chave, caso ele tenha optado por proteger a chave privada.

5.5.2 Parte 2

O resumo assinado na fase anterior será aqui decifrado com a chave pública do certificado de licença, que está contida no próprio certificado e então, comparado com um novo resumo do desafio gerado por um algoritmo de cálculo de resumo igual ao do estágio anterior. Se os resumos forem iguais, significa que o usuário possui acesso a chave privada correspondente ao par de chaves do certificado de licença e tem autorização para continuar com o processo de validação do software. Se a comparação dos dois resumos apresentar diferença, o processo de validação encerra-se imediatamente, bem como a execução do software protegido.

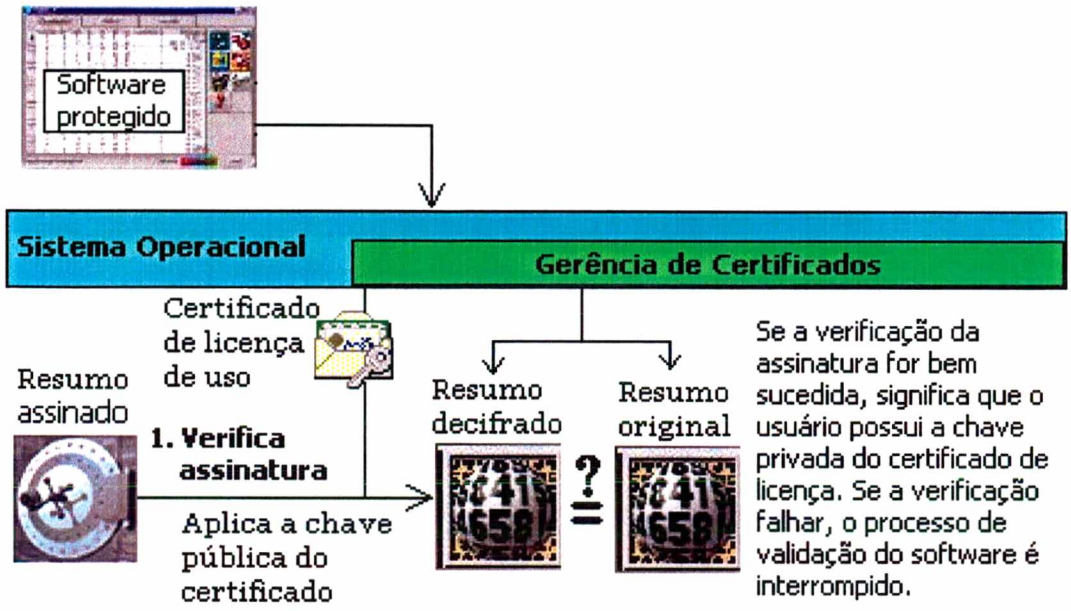

Figura 5.5: Processo de validação da licença de uso do software - parte 2. Nesta segunda parte, o software protegido solicita que a gerência de certificados aplique a chave pública contida no certificado de licença para verificar a assinatura do resumo assinado na primeira parte do processo.

A figura 5.5 ilustra esta fase de verificação da assinatura, que continua transparente para o usuário e executada pela gerência de certificados.

5.5.3 Parte 3

Chegar nesta etapa significa que o usuário tem acesso à chave privada do par de chaves do certificado de licença de uso do software.

Dessa maneira, o primeiro procedimento que o software protegido deverá fazer é verificar a data de validade do certificado de licença. Se a data expirou, todo o processo pára aqui. Caso contrário, o próximo passo é consultar a LCR local do sistema operacional. Uma vez que a LCR é ainda válida, é preciso verificar se o certificado de licença consta nesta lista. Se o certificado foi revogado, o processo de validação é interrompido. Estas operações são todas requisitadas pelo software

protegido e executadas pela gerência de certificados.

Por outro lado, se o certificado de licença não consta na LCR local do sistema operacional, efetua-se o próximo procedimento que é uma consulta a uma LCR remota [6] e mantida por uma AC. Se a consulta acusar que o certificado de licença foi revogado, interrompe-se todo o processo de validação. Se nada constar, o software protegido é liberado para uso.

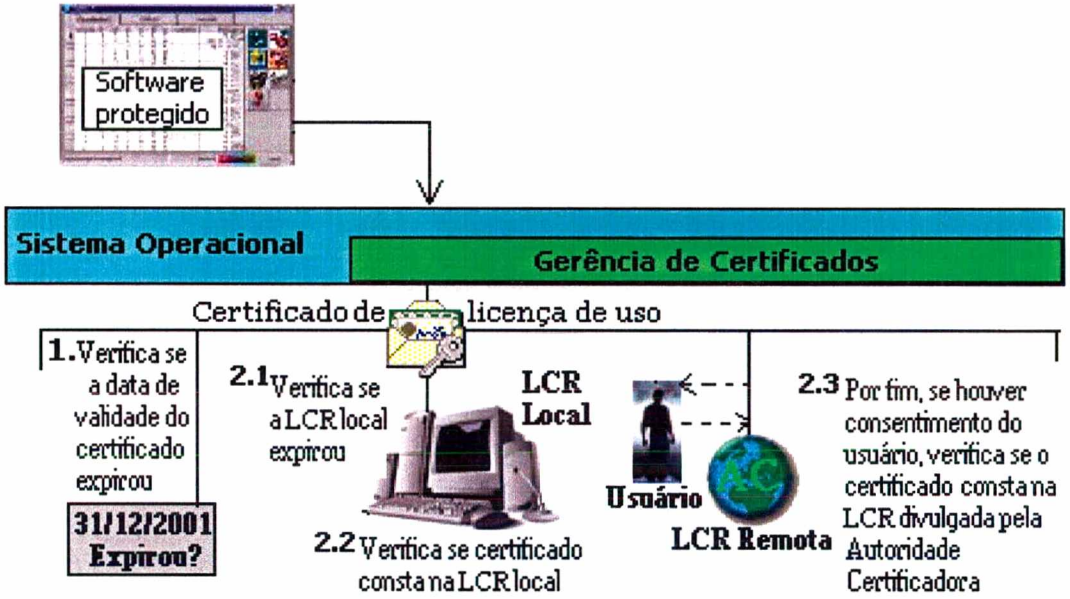

Figura 5.6: Processo de validação da licença de uso do software - parte 3. Na última parte do processo de validação é necessário verificar (através da gerência de certificados) se o certificado ainda é válido. Para isto, verifica-se: 1) se a validade do certificado não expirou; 2.1) se a validade da LCR local não expirou; 2.2) se o certificado não consta na LCR local e 2.3) com o consentimento do usuário, consultar se o certificado não consta em uma LCR remota.

Pela ilustração da figura 5.6 podemos ver os procedimentos finais de validação do software protegido: validação da data de expiração do certificado de licença e da data de expiração da LCR local do sistema operacional, bem como consulta local e remota da LCR. Vale observar que, desde o momento da instalação

[6] OCSP - Online Client Status Protocol [TAH 99] e [AUS 01].

do software protegido, pode-se deixar a opção de consulta remota à LCR configurável pelo usuário, preservando-se, assim, o direito à privacidade.

Apesar do Processo de Validação da Licença de Uso do Software ser um pouco extenso e aparentemente complicado, é bom lembrar que ele é transparente ao usuário e bastante rápido até o momento da consulta remota a uma LCR. Quanto à consulta remota a uma LCR, isto dependeria da banda disponível na rede.

5.6 Personalização do Software

Uma vez que a validação do software seja bem sucedida, pode-se personalizar o programa com as informações contidas no certificado de licença. Como o certificado é comprovadamente à prova de fraudes [FEG 99], as informações contidas nele são confiáveis, por isto, outras extensões contendo outros dados do usuário, como: endereço, data de nascimento, naturalidade, etc, poderiam ser adicionadas ao certificado para enriquecer a personalização do software.

5.7 Fraquezas e Limitações do Modelo

O que propomos com esta Dissertação é uma forma mais eficiênte de combate à pirataria. Sabemos que o uso de cópias ilegais de software é um mal difícil de ser combatido. Por esta razão, iremos expor neste item, alguns pontos que consideramos fraquezas ou limitações do Modelo. Porém, iremos falar também sobre alternativas para estas fraquezas e limitações:

computadores não conectados à Internet: existe o problema dos computadores que não estão conectados à Internet e que, em função disto, não propiciam uma consulta às listas de certificados revogados. Como alternativa a esta limitação, pode-se experar pela expiração do certificado;

alteração da data do computador: o usuário pode, também, tentar burlar a validação da data de expiração do certificado, alterando a data do computador

para uma data que esteja sempre dentro do período de validade do certificado. Em contrapartida, apesar da alteração da data do computador ser um processo simples de ser feito, ele pode se tornar cansativo com o tempo, além de comprometer outras aplicações que dependam diretamente da data, como: agendas eletrônicas, cálculo de juros e demais serviços;

usuário sofisticado: se o usuário do modelo for bastante sofisticado, obtiver recursos consideráveis, tempo, paciência e dominar o campo da engenharia reversa, ele pode quebrar o sistema de proteção e retirá-lo, gerando um novo "executável". Podemos dizer que este usuário iria encontrar mais dificuldades em retirar as proteções de um software protegido por certificação digital do que em qualquer outro esquema de proteção. Isto porque não é o software protegido que executa diretamente as operações de validação. Quem executa estas operações é o sistema operacional, atrávés da gerência de certificados (veja item 5.4). Ainda como alternativa a esta ameaça, a assinatura de código poderia ser usada para assinar este "executável", condicionando a sua execução somente quando ele se encontrar íntegro. Seria inviável computacionalmente para este usuário sofisticado, quebrar a criptografia do resumo gerado;

falência da AC: existe a possibilidade da autoridade certificadora a qual o produtor/revendedor mantenha convênio, vir a falir ou ser extinta. Neste caso, os serviços referentes aos certificados de licença correriam o risco de serem cessados. Na maioria dos países, existem regras rígidas para concessão de serviços de emissão de certificados digitais. Estas regras envolvem exigências como reservas financeiras para garantir as operações da AC e no caso de falência, a transferência dos clientes para uma outra AC. Ainda assim, com o intuito de proteger os direitos do consumidor, seria prudente existir uma cláusula no termo de contrato em que estabeleca a obrigação do produtor/revendedor em garantir a continuidade do serviço buscando alternativas;

falência do produtor: na hipótese do produtor do software vir a falir, seria pru-

dente existir uma cláusula no termo de contrato em que estabeleca a obrigação do produtor em fornecer certificados definitivos (sem expiração e sem ônus) para todos os seus clientes usuários do Modelo. Este tipo de atitude demonstra boa índole por parte do produtor e inspira confiança ao o cliente para que este aceite os termos do contrato.

5.8 Implementação do Modelo com Base na Construção de um Protótipo Didático

Neste item, o leitor terá uma idéia melhor de como o modelo de Proteção de Software por Certificação Digital pode ser implementado na prática.

A implementação é composta de alguns componentes que são essenciais para o funcionamento do modelo. Já de início, contamos com o apoio de uma AC que é responsável pela confecção dos certificados de licença de uso de software protegido. Estes certificados são baseados na recomendação X.509v3 e carregam com eles extensões específicas deste Projeto. A AC também é responsável pela manutenção da LCR. Para esta experiência foram emitidos dois certificados:

- um certificado válido [7];
- um certificado revogado.

Já a construção do protótipo, onde todo o modelo será testado, foi propositadamente dirigida para fins didáticos, onde todas as três partes de processo de validação da licença de uso do software estão bem definidas. A seguir, vamos discutir com maiores detalhes como o protótipo foi construído.

[7] Testes com a expiração da data de validade serão feitos com o certificado válido, alterando-se a data do computador.

5.8.1 Construção do Protótipo Didático

Conforme planejado e aprovado no Trabalho Individual [ROC 01], a linguagem de programação e sistema operacional escolhidos para implementar o modelo foram o Visual Basic® e o Windows®, ambos produtos da Microsoft®. Esta foi uma escolha pessoal do autor, o que não que dizer que seja a única opção. Alternativamente, uma outra linguagem poderia ser usada, como, por exemplo, o Java™ da Sun® Microsystems, utilizando as facilidades do keytool para gerenciamento de chaves e certificados. O ambiente operacional também poderia ser outro, como o Linux®.

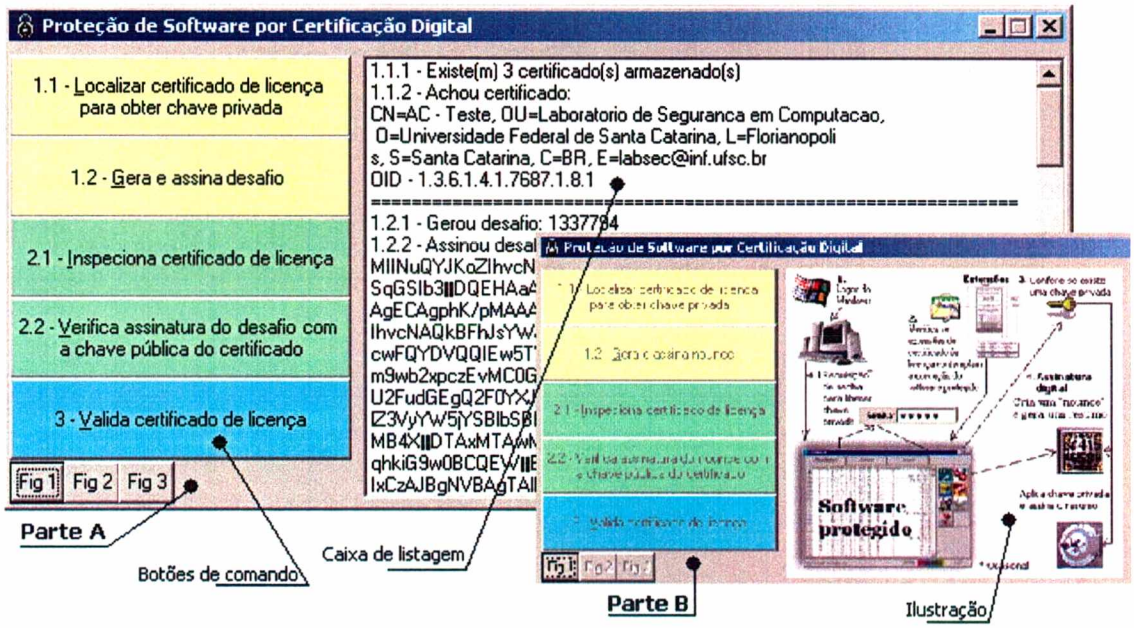

Figura 5.7: Tela do protótipo com a caixa de listagem (A) e ilustrações (B). Na parte A da figura podemos ver os botões de comando na seqüência: 1.1, 1.2, 2.1, 2.2 e 3. A caixa de listagem que se encontra na parte A, mostra as ações provocadas por cada um dos botões. Na parte B da figura, a caixa de listagem é substituída pelas ilustrações anteriormente apresentadas, que são: figuras 5.4, 5.5 e 5.6.

A interface do protótipo é constituída de uma única tela com cinco botões principais, numerados em seqüência, no intuito de separar as três etapas de

validação da licença de uso do software. No caso, os dois primeiros botões, 1.1 e 1.2, são responsáveis pela primeira parte da validação. Os dois botões seguintes, 2.1 e 2.2, são reponsáveis pela segunda parte da validação e o último botão, 3.1, é responsável pela última parte. Isto está ilustrado pela parte A da figura 5.7.

Além dos botões, existe uma caixa de listagem que serve para documentar as ações internas do protótipo. Para auxiliar nas explicações é possível mostrar as figuras: 5.4, 5.5 e 5.6 (páginas 57, 58 e 59) durante a execução do protótipo, conforme ilustra a parte B da figura 5.7.

5.8.2 Parte 1 da Implementação

A CryptoAPI (veja item 3.8) teve uma importância fundamental na implementação do modelo. A Capicom.dll [COR 01a], um controle "activex" desenvolvido pela Microsoft® e que "empacota" algumas funções da CryptoAPI, foi utilizado como referência no projeto do protótipo em Visual Basic®.

A primeira parte de validação da licença de uso do software protegido é representada pelos dois primeiros botões do protótipo. O primeiro, botão 1.1, contém ações que devem ser executadas pelo software protegido assim que este for acionado pelo usuário. Considerando que o usuário esteja com o sistema operacional carregado e "logado" em sua sessão e que já esteja com o certificado de licença devidamente importado para sua pasta de certificados (estes são procedimentos típicos do sistema operacional), o software após ser acionado deve:

- **Localizar o Certificado de Licença de Uso do Software:**
  O usuário pode abrigar em seu sistema, certificados de diversos tipos e propósitos. O software protegido deve procurar, nesta coleção, o certificado que possua uma extensão que indique que o seu uso é para validação de software protegido por certificação digital. Para esta experiência foi utilizado o código de OID do Projeto como indicador desta extensão. A seguir, um exemplo como poderia ser este código em Visual Basic®:

Set Store = CreateObject("CAPICOM.Store")

:::

For Each Certificate In Store.Certificates

If Certificate.ExtendedKeyUsage.EKUs(0).OID = " 1.3.6.1.4.1.7687.1.8.1"Then

:::

- **Verificar se o Certificado de Licença de Uso do Software possui chave privada associada:**
  Após selecionar o certificado certo na coleção, o próximo passo é verificar se o usuário possui uma chave privada correspondente ao par de chaves do certificado. Então, poderíamos programar do seguinte modo:

  :::

  If Certificate.HasPrivateKey Then

  :::

É no botão 1.2 que o desafio é lançado para o usuário. Gera e assina-se um desafio com a chave privada recém localizada.

- **Geração do desafio:**
  Uma operação simples, que gera um número identificador único, feita em Visual Basic®:

  Dim num As Double Static x As Long

  :::

  desafio = Format(Date,"dd") & Format (Time, "hhmmss")

  num = Val(desafio)

  Randomize num

  desafio = Format((Int(Rnd * num)), "############0")

  :::

- **Assinatura do desafio:**

  Para assinar usa-se a chave privada correspondente. O algoritmo usado para gerar o resumo é o SHA-1 com saída de tamanho de 160 bits. Para cifrar este resumo, usa-se o algoritmo RSA.

  Set Signer = CreateObject("CAPICOM.Signer")

  :::

  Signer.Certificate = Certificate

  :::

  Signeddata.Content = desafio

  Signature = Signeddata.Sign(Signer, False,

  capicom.CAPICOM_ENCRYPTION_KEY_LENGTH_MAXIMUM)

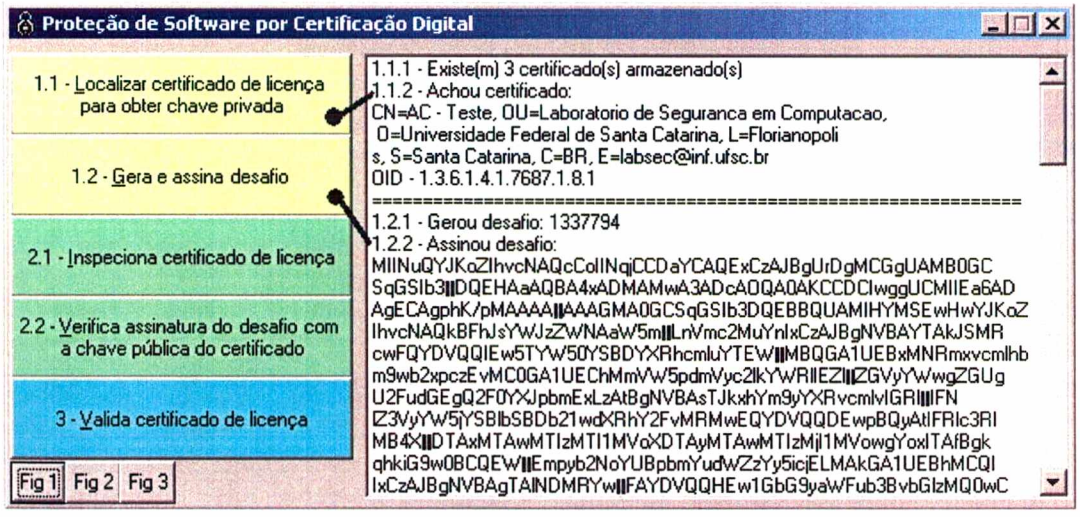

Figura 5.8: Parte 1 da Implementação. O botão 1.1 quando acionado, realiza a localização do certificado apropriado para este modelo de proteção e ainda exibe suas características. O botão 1.2, gera e assina o desafio.

As operações desta primeira parte estão ilustradas na figura 5.8.

5.8.3 Parte 2 da Implementação

A segunda parte da validação começa com o botão 2.1. Este botão apenas inspeciona o certificado de licença, verificando a sua assinatura. Na verdade esta operação poderia ser suprida, pois se o certificado estivesse violado ele nem apareceria na lista de coleção de certificados, já que a própria capicom.dll iria rejeitá-lo. Já o botão 2.2 realiza uma operação importante de verificação da assinatura do desafio.

- **Inspeção do certificado:**
  O certificado é assinado por uma AC. Desta forma, ele pode ser facilmente validado:

```
Dim chain As New chain
Dim cert As New Certificate
:::
Set chain = CreateObject("CAPICOM.Chain")
Set Store = CreateObject("CAPICOM.Store")
:::
Set cert = Certificate
cert.IsValid.CheckFlag = CAPICOM_CHECK_SIGNATURE_VALIDITY
If cert.IsValid.Result Then
'OK
Else
chain.Build cert
If CAPICOM_TRUST_IS_NOT_SIGNATURE_VALID And chain.Status Then
ind = ind + 1
texto(ind) = texto(ind) & "INSPEÇÃO FALHOU - Status = "
& _ chain.Status & " Name = "& _
cert.GetInfo(CAPICOM_CERT_INFO_SUBJECT_SIMPLE_NAME)
:::
```

- **Verificação da assinatura do desafio:**

  Como o desafio foi anteriormente assinado com uma chave privada, podemos verificar esta assinatura com a chave pública contida no certificado de licença para termos certeza de que o usuário possui a chave privada correspondente ao par de chaves do certificado.

  Dim sig As Signeddata

  Set sig = New Signeddata

  :::

  sig.Content = desafio

  :::

  sig.Verify Signature, False, CAPICOM_VERIFY_SIGNATURE_ONLY

  :::

Figura 5.9: Parte 2 da Implementação. O botão 2.1, quando acionado, inspeciona o certificado para saber se não houve nenhuma tentativa de violação. O botão 2.2 verifica a assinatura do desafio, aplicando, agora, a chave pública contida no certificado.

As operações desta segunda parte estão devidamente ilustradas na figura 5.9.

5.8.4 Parte 3 da Implementação

A última parte da validação, representada pelo botão 3.1, é responsável pela validação do certificado de licença, no que se refere a prazo de validade e inclusão em listas de revogação.

- **Prazo de validade do certificado:**

  O certificado X509v3 possui um período de validade que vai, por exemplo, de d1 até d2. Se a data atual for menor que d1 ou maior que d2, o certificado é considerado fora de uso:

```
Dim chain As New chain
Dim cert As New Certificate

Set Store = CreateObject("CAPICOM.Store")
Set chain = CreateObject("CAPICOM.Chain")
:::
Set cert = Certificate
cert.IsValid.CheckFlag = CAPICOM_CHECK_TIME_VALIDITY
If cert.IsValid.Result Then 'OK
Else
chain.Build cert
If CAPICOM_TRUST_IS_NOT_TIME_VALID And chain.Status Then
ind = ind + 1
texto(ind) = texto(ind) & "EXPIRADO - Status = "& _
chain.Status & " Name = "& _
cert.GetInfo(CAPICOM_CERT_INFO_SUBJECT_SIMPL_NAME)
End If
End If :::
```

- **Revogação do certificado:**

Figura 5.10: Parte 3 da Implementação. O botão 3, quando acionado, verifica primeiro se o certificado não se encontra expirado e, depois, realiza consultas em LCRs locais e remotas para ver se o certificado não foi revogado.

As Listas de Certificados Revogados e validade da listas são automaticamente verificadas pela CryptoAPI. Se nada constar na lista local, então a pesquisa parte para a consulta remota, caso haja conexão com a Internet:

```
Set Store = CreateObject("CAPICOM.Store")
Set chain = CreateObject("CAPICOM.Chain")
:::
Set cert = Certificate
cert.IsValid.CheckFlag = CAPICOM_CHECK_ONLINE_REVOCATION_STATUS
If cert.IsValid.Result Then
'OK
Else
ind = ind + 1
chain.Build cert
If CAPICOM_TRUST_IS_REVOKED And chain.Status Then
texto(ind) = texto(ind) & "REVOGADO - Status = "& _
```

```
chain.Status & " Name = "& _
cert.GetInfo(CAPICOM_CERT_INFO_SUBJECT_SIMPLE_NAME)
End If
:::
```

Veja a ilustração da figura 5.10 para acompanhar esta última parte da validação.

5.9 Conclusão

Neste capítulo constatamos que o modelo de Proteção de Software por Certificação Digital tem sua fundamentação baseada em uma infra-estrutura de chave pública e que, por isto, proporciona ao produtor um maior controle sobre as cópias piratas. Foi apresentado, também, como foi realizada a experiência de implementação do modelo, descrevendo todas as partes de validação da licença de uso do software.

Como foi citado, algumas fraquezas existem no modelo, mas estas podem ser compensadas com as alternativas expostas aqui neste capítulo. Não obstante, a ligação entre usuário, certificado digital e software, criam um compromisso sério e eficiente, que faz com que o usuário repense em quebrar com o termo de contrato do software.

Capítulo 6

Considerações Finais

6.1 Introdução

Neste capítulo final, os objetivos atingidos na construção de um Modelo de proteção de software baseado em certificação digital serão revistos no item 6.2. No item 6.3, destacaremos a eficácia do Modelo. Já no item 6.4, abordaremos trabalhos futuros que podem ser realizados para aperfeiçoar o Modelo em sua aplicação comercial e como conseqüência desta aplicação comercial, benefícios indiretos podem ser gerados para a sociedade. Falaremos destes benefícios no item 6.5. Por último, no item 6.6, a dissertação será concluída.

6.2 Objetivos Atingidos

Os principais objetivos desta dissertação foram traçados bem no início, logo no capítulo 1. Por se tratar de uma pesquisa para construção de um modelo de proteção de software que ainda não havia sido concebido antes, a motivação foi fator decisivo para vencer os desafios e as dificuldades que surgiram no decorrer do trabalho. Os objetivos desta pesquisa foram concluídos com êxito em sua totalidade:

- estudo de material técnico-científico necessário para a formação da teoria

de um Modelo de Proteção de software, bem como fornecer embasamento e retórica ao texto da dissertação;

- estudo do ambiente LaTeX [1] para prover qualidade ao relatório técnico-científico da dissertação;
- concepção de um Modelo de Proteção de Software por Certificação Digital que pudesse oferecer vantagens como: ligação usuário-software-certificado, neutralização das cópias piratas através da revogação do certificado, concessão de direito de uso do software por tempo determinado e personalização do software;
- construção e implementação de um protótipo para testes do Modelo;
- obtenção dos certificados digitais baseados na recomendação X.509v3, com as extensões necessárias para o teste do protótipo;
- análise de resultados com base em pesquisa de campo realizada.

6.3 Eficácia da Proteção de Software por Certificação Digital

Este trabalho se põe à frente de seu tempo por propor uma idéia inovadora. Diferentemente de outras formas de proteção já criadas e que se concentram em tentar impedir que a cópia física do software seja feita, a Proteção de Software por Certificação Digital une o software ao usuário através de um certificado digital. Esta ligação traz vantagens para o produtor de software, são elas:

a) **ligação usuário-software-certificado:** esta união é inegável por parte do usuário, uma vez que este faça a requisição do certificado de licença, ele tem como pré-requisito, aceitar todas as condições do termo de contrato de uso do software,

[1] LaTeX é um conjunto de macros para o TeX (programa para edição de textos científicos de excelente apresentação gráfica).

o que o torna consciente das penalidades cabíveis em caso de quebra do contrato. Somente o usuário possui a chave privada a ser usada com o certificado. Isto torna bem mais fácil comprovar a participação do usuário mal intencionado no processo que desencadeia a pirataria. O certificado de licença é garantido por uma autoridade certificadora e é comprovadamente à prova de fraudes [FEG 99]. Ele tem a finalidade específica de ser usado para o software em questão;

b) **neutralização das cópias piratas pela revogação do certificado:** através de auditorias contratadas ou em parceria com órgãos de combate à pirataria, o produtor de software pode identificar os certificados que estão sendo distribuídos com as cópias ilegais. Deste modo, o produtor pode solicitar a uma autoridade certificadora que revogue os certificados de licença usados com estas cópias piratas, desativando, assim, o uso delas no ato de validação do software;

c) **concessão de direito de uso do software por tempo determinado:** como o certificado possui um período pré-estabelecido para seu uso e não pode ser usado antes e nem depois deste período, fica mais fácil para o produtor controlar a validade das concessões de licenças para uso do software, tendo a possibilidade de substituir a venda pelo aluguel;

d) **personalização do software:** os certificados digitais tem a vantagem de trazer informações a respeito de seus proprietários. Estas informações podem ser aproveitadas para personalizar o software.

6.4 Trabalhos Futuros

O trabalho e a pesquisa para aperfeiçoamento do Modelo de Proteção de Software por Certificação Digital devem continuar. Há ainda muito a ser feito até conseguirmos viabilizar a proteção por certificação digital. Muito do que falta fazer é relativo a outras áreas de pesquisa e, portanto, fogem ao escopo deste

trabalho de dissertação. Porém, algumas sugestões podem ser mencionadas. São elas:

a) **gestão empresarial:** pesquisas no campo da Administração devem ser conduzidas para conceber um modelo ideal de gestão empresarial que atenda as necessidades do Modelo de Proteção. Além disto, para uma maior eficácia e/ou viabilidade comercial, deve-se estudar a possibilidade de se implementar o modelo em conjunto com outros métodos de proteção, como, por exemplo, o modelo visto no item 2.3.1;

b) **termo de contrato:** a questão jurídica que envolve o termo de contrato entre produtor e usuário também deve ser alvo de pesquisa no campo do direito, respeitando, claro, o lado do consumidor, mas provendo termos legais e eficazes que reforcem a aliança entre o usuário, software e certificados;

c) **robustez:** a capacidade de o Modelo ser resistente a ataques, principalmente os de engenharia reversa, deve ser posta à prova. Para efetuar este teste, desafios devem ser lançados à comunidade de "crackers" e de engenharia reversa, solicitando a quebra do código e retirada da proteção ou fraudação do esquema de validação. Porém, este desafio deve ser bem organizado. Um novo protótipo deve ser produzido, pois o atual é bastante óbvio no que tange aos procedimentos que estão sendo executados. O código final do novo protótipo deve ser assinado através de uma ferramenta de assinatura de código, como o authenticode da Microsot®, com um certificado emitido por uma AC conveniada. As tarefas a serem seguidas pelos desafiantes devem ser bem estruturadas e deve-se exigir prova convincente de objetivo atingido;

d) **módulo de denúncia:** como alternativa de redução de custo com a contratação de auditorias para descoberta de cópias ilegais, a construção de um módulo de denúncias a ser implementado junto com o Modelo poderia ser feita. Qualquer usuário que suspeite estar usando uma cópia pirata poderia efetuar denúncia através de correspondência eletrônica, sem, necessariamente, se identificar.

A grande vantagem deste módulo é que a mensagem poderia ser facilmente assinada com a respectiva chave privada, e que por isto, daria autenticidade a esta denúncia. Desta maneira, o produtor teria fortes razões para colocar o certificado de licença denunciado, sob fortes suspeitas.

6.5 Benefícios Indiretos

A Proteção de Software por Certificação Digital se for bem implementada e obtiver sucesso comercial, pode vir a reduzir em muito a pirataria praticada. De imediato, isto iria aumentar a arrecadação das empresas produtoras de software e, conseqüentemente, a arrecadação dos impostos obtidos pelas vendas. Pode ganhar também o consumidor, pois se houver um número bem maior de usuários pagantes para determinado software, este pode ter seu preço reduzido. Além disso, as Autoridades Certificadoras teriam um novo nicho em potencial para a emissão de certificados destinados a validação de licença de software. As ACs também se beneficiariam com contratos de exclusividade para emissão de certificados de licença mantidos com grandes produtores de software. Todos estes benefícios, proporcionados por um modelo de proteção mais eficiente, poderiam aquecer a economia e gerar muitos empregos.

6.6 Conclusão Final

O combate à pirataria sempre foi um trabalho árduo e ineficaz para os desenvolvedores de software. Apesar dos esforços feitos para se proteger o software, ainda não foi encontrado uma forma eficiente de se fazer esta proteção. De acordo com o relatório de 1999 da SIIA [2], a pirataria ainda lesa as empresas e desenvolvedores de software em todo o mundo, na ordem de bilhões de dólares por ano.

A certificação digital é hoje uma realidade que conta com a força

[2]SIIA – Software & Information Industry Association [ASS 00]

da criptografia como forma de assegurar a inviolabilidade do certificado. Existe ainda todo um padrão bem elaborado de emissão de certificados, hierarquias entre ACs e validação de caminhos de certificação que dão suporte a certificação digital, assegurando um futuro certo de crescimento e aceitação global deste novo nicho de mercado. O amparo legal também tem acompanhado esta evolução, já sendo presente na legislação de muitos países. Isto garante uma grande importância ao certificado digital como instrumento de prova em operações comerciais pela Internet.

Conciliar o certificado digital à venda de software é apostar numa proteção que tem grandes chances em dar certo. O usuário comprador certamente não iria gostar de assumir as penalidades resultantes em liberar a chave secreta de seu certificado para piratear o software. Este encargo de responsabilidade, além de sensibilizar a consciência do usuário, é prova evidente da procedência da licença do software.

Referências Bibliográficas

[AN 01] ANDREW NASH, WILLIAM DUANE, C. J. **PKI: Implementing and Managing E-Security**. Berkeley, CA, EUA:McGraw-Hill, 2001.

[ASS 00] ASSOCIATION, S. S. . I. I. Siia's report on global software piracy 2000. SIIA Software & Information Industry Association, EUA, 2000. Relatório técnico.

[AUS 01] AUSTIN, T. **PKI. A Wiley Tech Brief**. New York, NY, EUA:John Wiley & Sons, Inc., 2001.

[BRA 98] BRASIL. Lei número 9.606, de 19 de fevereiro de 1998, lei de software. Congresso Nacional, 1998.

[CHO 95] CHOUDHURY, A. K. et al. Copyright protection for electronic publishing over computer networks. **IEEE Network Magazine**, [S.l.], v.9, n.3, 1995.

[COR 96a] CORPORATION, M. **Application Programmer's Guide**. Redmond, WA, EUA:Microsoft CryptoAPI Preliminary, 1996. disponível em: http://www.graphcomp.com/info/specs/ms/capi.html, acessado em: 03/07/2001.

[COR 96b] CORPORATION, M. Ensuring accountability and authenticity for software components on the internet. **Microsoft Authenticode Tecnology, Redmond, WA, EUA**, [S.l.], v., 1996.

[COR 01a] CORPORATION, M. Microsoft msdn library - capicom. **disponível em: http://msdn.microsoft.com, acessado em: 26/10/2001**, [S.l.], v., 2001.

[COR 01b] CORPORATION, M. Microsoft msdn library - cryptoapi. **disponível em: http://msdn.microsoft.com, acessado em: 26/10/2001**, [S.l.], v., 2001.

[dR 01] DE REZENDE, P. A. D. Onde estão os verdadeiros crimes de informática? **disponível em: http://www.cic.unb.br/docentes/pedro, acessado em: 07/11/2001**, [S.l.], v., 2001.

[EAS 00] EAST, R. L. **PKCS #10 v1.7: Certification Request Syntax**. Massachusetts, MA, EUA:RSA Laboratories, 2000.

[FEG 99] FEGHHI, J.; FEGHHI, J. **Digital Certificates - Applied Internet Security**. Massachusetts, MA, EUA:Addison Wesley Longman, Inc., 1999.

[FOR 97] FORD, W. **Secure Electronic Commerce**. New Jersey, NJ, EUA:Prentice Hall PTR, 1997.

[GAR 99] GARFINKEL, S. L. **Comércio e Segurança na WEB. Riscos, Tecnologias e Estratégias**. São Paulo, Brasil:Market Books do Brasil, 1a ed. ed., 1999.

[KAL 97] KALISKI, B. **Extensions and Revisions to PKCS #7 v 1.6**. Massachusetts, MA, EUA:RSA Laboratories East, 1997.

[KAL 98a] KALISKI, B. **PKCS #10 v 1.5: Certification Request Syntax**. Massachusetts, MA, EUA:RSA Laboratories East, 1998.

[KAL 98b] KALISKI, B. **PKCS #7 v 1.5: Cryptographic Message Syntax**. Massachusetts, MA, EUA:RSA Laboratories East, 1998.

[oRE 97] OF REVERSE ENGINEERING, H. A. Project 7: Most stupid protections. **+HCU: Academy of Reverse Engineering**, [S.l.], v., 1997.

[ROC 01] ROCHA, J. L. F. Proteção de software por certifiação digital - trabalho individual. Univesidade Federal de Santa Catarina, Brasil, 2001. Relatório técnico.

[STA 99] STALLINGS, W. **Cryptography and Network Security. Principles and Practice**. New Jersey, NJ, EUA:Prentice-Hall Inc, 1999.

[TAH 99] TIMOTHY A. HOWES, M. C. S. **Understanding and Deploying LDAP Directory Services**. EUA:MTP, Macmillan Tecnical Publishing, 1999.

[UNI 97a] UNION, I.-T. I. T. Recommendation x.500 (1997) — isso/tec 9594-1:1997 information technology - open systems interconnection - the directory: Overview of concepts, models and services. ITU-T International Telecomunication Union, 1997. Relatório técnico.

[UNI 97b] UNION, I.-T. I. T. Recommendation x.509 (1997) — isso/tec 9594-8:1993, information technology - open systems interconnection - the directory: Authentication framework. ITU-T International Telecomunication Union, 1997. Relatório técnico.

[WIN 99] WINER, E. Copy protection - the audio industry's dirty litlle secret. **PROREC, EUA**, [S.l.], v., 1999.

[YH 99] Y.L. HUANG, S.P. SHIEH, F. H. A generic electronic payment model supporting multiple merchant transactions. **Computers and Security**, [S.l.], v., 1999.

Bibliografia Adicional

Es97 Dino Esposito, "Supporting CryptoApi in Real-World Applications", Microsoft Interactive Developer, EUA, 1997, disponível em: http://www.microsoft.com/Mind/0697/CRYPTO.HTM, acessado em: 19/09/2000.

FlOl94 Lúcia Locatelli Flôres, Lúcia Maria Nassib Olímpio e Natália Lobor Cancelier, "Redação. O Texto Técnico/Científico e o Texto Literário", Editora da UFSC, Florianópolis, SC, 1994.

PR00b Pedro Antonio Dourado de Rezende, "Jon Johansen: Bandido ou Herói?", Brasil, 2000, disponível em: http://www.cic.unb.br/docentes/pedro - acessado em: 07/11/2001.

PR00c Pedro Antonio Dourado de Rezende, "Kafka, Orwell e crimes digitais", Brasil, 2000, disponível em: http://www.cic.unb.br/docentes/pedro - acessado em: 07/11/2001.

PR01d Pedro Antonio Dourado de Rezende, "Sapos Piramidais nas Guerras Virtuais. Paradoxos da Propriedade Intelectual e da Segurança Computacional", Brasil, 2001, disponível em: http://www.cic.unb.br/docentes/pedro - acessado em: 07/11/2001.

Bo00 Richard Bondi, "WCCO 1.0 Manual", John Wiley & Sons, Inc., New York, NY, EUA, 2000.

WIPO96 U.S. Copyright Office, "WIPO Copyright Treaty", United States Copyright Office - The Library of Congress, EUA, 1996, disponível em:
http://lcweb.loc.gov/copyright/wipo,
acessado em: 15/07/2000.

Glossário

A

AC, autoridade certificadora - Uma entidade de confiança que emite identidades digitais requeridas por pessoas físicas, jurídicas e computadores.

Activex, Activex Control - Um tipo de ambiente para aplicações da Microsoft® que inclui um grande número de componentes que operam e/ou ligam diferentes aplicações em um computador ou rede de computadores.

Algoritmo assimétrico - algoritmo usado por cifradores que utilizam pares de chave: chave pública/privada. Enquanto uma chave é usada para cifrar, a outra é usada para decifrar.

Algoritmo simétrico - algoritmo usado por cifradores que utilizam uma chave secreta para cifrar. São mais rápidos do que os algoritmos assimétricos.

Assinatura digital - Um resumo cifrado por uma chave privada e usado para autenticar mensagens ou arquivos e garantir a integridade deles.

B

BIT - Algarismo binário.

Buffer - Área usada para armazenamento temporário de dados na memória do computador.

C

Cd, cd(s) - Abreviatura de "compact disk". Mídia ótica usada para armazenar arquivos, dados ou programas.

Cifrador - Programa que contém um algoritmo usado para cifrar mensagens ou arquivos, geralmente utilizando chaves pública/privada ou uma chave secreta.

Cracker(s) - Definição adaptada para o português brasileiro falado e que define o usuário sofisticado com capacidade de "retirar"proteções de cópia de programas de computador.

Crakeado(s) - Uma versão de um programa "aberto"por cracker(s).

D

Decompilador - Programa, geralmente usado por usuários sofisticados, para obter o resultado inverso da compilação, ou seja, transformar linguagem de máquina em código fonte. O processo original é feito por compiladores.

Demo - Veja shareware.

Desafio - Número aleatório, de difícil repetição, gerado por um algoritmo.

DLL, Dynamic-link Lybraries - Arquivo que contém rotinas do sistema operacional Windows®.

Download - Processo de baixa de arquivos pela Internet.

E

Engenharia reversa - Arte ou ciência estudada e praticada por usuários sofisticados que tem o intuito de descobrir como funciona um programa ou dispositivo em particular.

F

Firmware - Um hardware com circuitos pré programados para realizar tarefas específicas.

Freeware - Programa de computador distribuído de graça.

H

Hardware - Periférico, computador.

I

Interface - Método de comunicação entre duas partes.

L

Login, Logon, Logado - Procedimento para se criar a primeira conexão a um sistema operacional. Quando um usuário já criou esta conexão, fala-se que ele está "logado".

LCR, Lista de Certificados Revogados - Um documento mantido e publicado por uma AC que lista certificados que não são mais válidos.

P

Par de chaves pública/privada - É um par de chaves pertencentes a um único usuário e que é utilizada na criptografia de chave pública. Enquanto a chave pública é livremente distribuída, a chave privada permanece exclusivamente com o dono.

R

Resumo - Um conjunto de caracteres mapeado de uma mensagem ou arquivo por uma função resumo e que é único. Se a mensagem ou arquivo sofre alterações, o resumo já não será o mesmo. Geralmente usado em assinaturas digitais para garantir a integridade do objeto.

S

SET - Secure Electronic Transaction - é um famoso protocolo de pagamento proposto pela VISA e pela MasterCard.

Shareware - Programa de computador em versão de demonstração e que expira depois de algum tempo de uso ou possui algumas funções desabilitadas. Também chamado de Trial ou Demo.

T

Trial - Veja shareware.

Software - Programa de computador.

X

X.509v3 - Recomendação internacional emitida pela ITU-T para confecção de certificados digitais.

Apêndice

A. Ensaio Preliminar de Estudo de Campo

Embora sem muita significância em termos quantitativos, foi realizado um ensaio preliminar de estudo de campo voltado para dois tipos distintos de usuário: o usuário final, que utiliza o software, e a empresa, que produz e/ou revende o software. Este ensaio foi realizado unicamente com o intuito de se ter uma idéia sobre a receptividade e aceitação do modelo. A seguir, a descrição de como foi aplicado o estudo de campo, bem como, uma exposição da análise dos resultados obtidos.

A.1. Ensaio Preliminar Estudo de Campo para o Usuário Final

Procurou-se aplicar questões de aspecto geral para este tipo de usuário, onde assuntos como: prática, origem e controle da pirataria foram abordados, bem como, o nível de conhecimento que o usuário comum tem sobre alguns componentes de infra-estrutura de chave pública e qual seria a aceitação de uma proteção que envolva a certificação digital. A seguir uma relação dos resultados obtidos:

- **Perfil do usuário entrevistado:**

  1. escolaridade: a grande maioria dos entrevistados possui nível superior completo. O nível de escolaridade mais baixo foi de segundo grau com-

pleto.

2. grau de utilização do computador: todos os entrevistados mantém contato diário com o computador, seja no trabalho ou em casa.

- **Número de usuários entrevistados:** 10

- **Resultados obtidos quanto a prática, origem e controle da pirataria:** Nesta primeira parte do questionário foi perguntado ao usuário final se, em algum momento, ele praticou ou possuiu programas piratas, qual razão predominante o levou a fazer isto e qual a fonte mais propícia para se obter cópias de programa pirata. A grande maioria dos entrevistados confessaram ter praticado a pirataria de programas de computador. Quando indagados sobre as possíveis razões que levam os usuários de computador a copiar ilegalmente programas, 50% dos entrevistados, disseram que praticam a pirataria por causa do alto preço do software, 33% disseram que a pirataria é praticada porque não existem meios eficientes de combate a este crime e 17% optaram em dizer que a pirataria ocorre porque não há punição para os infratores.

  O resultado obtido com as respostas sobre qual a principal fonte de obtenção de programas piratas para o usuário final, revelou que vinte e cinco por cento dos entrevistados afirmaram obter as cópias de programas piratas através da Internet. Dezessete por cento alegaram obter "pacotes"de cds oferecidos na Internet, jornais e revistas. Cinqüenta e oito por cento disseram que é através de amigos que conseguem copiar ilegalmente os programas.

- **Resultados obtidos quanto a eficácia no combate atual da pirataria:** A opinião quanto a tentativa atual de combate à pirataria também foi alvo de questionamento para o entrevistado. Neste tópico, foi perguntado ao entrevistado se ele tinha conhecimento de alguém que tenha sido obrigado a comprar um software por não achar uma cópia pirata deste em nenhum lugar. Cinqüenta por cento das respostas obtidas indicam que não houve conhecimento de nenhum caso deste tipo. Ou seja, ninguém nunca precisou comprar

um software por não haver uma versão pirata disponível no mercado. Quarenta por cento dos entrevistados disseram que já souberam de um software que ninguém tenha conseguido copiar ainda, mas que era só esperar um pouco que logo apareceria uma cópia pirata no mercado. Dez por cento souberam de casos onde o usuário precisou comprar uma cópia legal por não ter encontrado uma cópia pirata no mercado.

Como foi visto, uma das principais causas da pirataria, segundo os próprios usuários, é o alto preço do software. Mediante isto, é natural pensar que uma das formas mais eficientes para desestimular a pirataria seria reduzir o preço do software para o usuário final. Quando indagado sobre uma possível redução drástica no preço dos programas, porém ainda com a possibilidade de obtê-los gratuitamente, mesmo que de modo ilegal, pela Internet, os entrevistados responderam que na maioria (80%) passariam a comprar cópias de programas legalizadas e apenas 20% continuariam a obter cópias ilegais.

- **Resultados obtidos quanto ao nível de conhecimento do entrevistado com relação aos componentes de infra-estrutura de chave pública:**

  Quando indagado se conhecia o assunto sobre componente de infra-estrutura de chave pública, como certificado digital, autoridade certificadora, chave privada, etc., obteve-se um empate entre entrevistados com bom conhecimento sobre o assunto e entrevistados com conhecimento razoável.

- **Resultados obtidos quanto a aceitação do modelo de Proteção de Software por Certificação Digital:**

  Nesta parte do questionário foi criado, numa primeira instância, o desejo do usuário em adquirir uma licença para um software que teve o seu período de demonstração encerrado. Para isto, seria necessária a aquisição de um certificado digital que iria habilitar todas as funções do programa em questão, bem como a aceitação dos termos de compromisso do software. Dos entrevistados, 80% foram favoráveis ao modelo de Proteção de Software por Certificação

Digital, porém, 20% encontraram neste modelo, um fator complicador.

Na pergunta seguinte foi revelado ao entrevistado que o software comprado nestas condições seria difícil de ser copiado ilegalmente, sem que o certificado digital e a respectiva chave privada fossem fornecidos juntos com a cópia pirata. Foi revelado, também, que, ao fazer isto, o usuário poderia ser facilmente processado pelo produtor de software, sem ter como negar o crime. Com base na mudança do panorama, 87% dos entrevistados mantiveram a decisão anterior, conscientes agora de que não poderiam emprestar e/ou fornecer o certificado digital e a chave privada pra ninguém. Apenas 13% dos entrevistados mudaram de idéia por saber que não poderiam piratear o software sem arcar com conseqüências graves.

- **Anonimato:**

  Os resultados foram analisados em nível genérico, com o intuito de facilitar o estudo de campo e também de preservar a origem das respostas obtidas, uma vez que o assunto pirataria, por vezes, invoca a ilegalidade. Por isto mesmo, não há qualquer identificação dos entrevistados nos questionários aplicados.

- **Conclusão:**

  Como conclusão deste questionário aplicado ao usuário final, podemos dizer que alguns fatores antes comentados nesta dissertação (veja capítulo 2), como por exemplo, razões da prática da pirataria, ineficiência das tentativas de controle atuais, a não degradação do meio digital como fator decisivo de propagação e a formação de uma cultura pró-pirataria na comunidade de usuários, foram evidenciados nesta pesquisa. Por outro lado, confirmando que o computador está cada vez mais presente no dia-a-dia das pessoas, a maioria dos entrevistados mostrou algum conhecimento sobre temas como certificação e assinatura digital. Demonstraram, também, na grande maioria, uma propensão em aceitar o modelo de Proteção de Software por Certificação Digital.

A.2. Ensaio Preliminar de Estudo de Campo para Empresas

Questões sobre o Modelo de Proteção de Software por Certificação Digital, visando agora o aspecto empresarial, foram aplicadas por meio de um questionário para um grupo de duas empresas de informática situadas em Florianópolis, Santa Catarina. Também foi demonstrado, para estas empresas, o protótipo do Modelo com simulações de validação de certificados revogados e expirados. A seguir, a relação das empresas entrevistadas e os resultados obtidos.

- **Empresas entrevistadas:**

1. **3lj Criação e Desenvolvimento**
   Endereço: Rua dos Ilhéus 46, sl: 501
   Centro, Florianópolis - SC
   CEP 88010-560
   Telefone: (0xx48) 322-0699
   Endereço eletrônico: www.3lj.com.br
   Representante: Luiz Otávio Borrajo Costa
   Cargo: Diretor-Presidente
   Ramo de atividade: Prestadora de pequenos serviços em informática e criação em desenvolvimento WEB.

2. **APRESI**
   Endereço: Trv. Ratcliff, 25 - S/ 204
   Hotel Royal - Centro
   CEP: 88010-420
   Florianópolis - SC
   Telefone: (0xx48) 223-7213
   Endereço eletrônico: www.apresi.com
   Representante: Wallace da Silva Pereira
   Cargo: Diretor Técnico Operacional
   Ramo de atividade: Prestadora de pequenos serviços em informática e

produtora de software.

- **Resultados obtidos quanto a experiência com problemas causados pela pirataria:**

  As empresas entrevistadas foram unânimes em relatar que já vivenciaram problemas que envolvam cópias ilegais de seus sistemas. Frente a isto, foi revelado que algumas tentativas foram feitas para tentar evitar prejuízos. Entre as opções escolhidas, estão: o uso de número de série, contra-senha e disquete de proteção. Das tentativas realizadas nenhuma das empresas se mostrou 100% satisfeita com o desempenho e nível de proteção proporcionado pelas formas de proteção de software utilizadas. Em especial, uma das empresas revelou ter tido problemas com disquetes de proteção com defeito.

- **Resultados obtidos quanto a possibilidade de experimentar o Modelo de Proteção de Software por Certificação Digital:**

  A pesquisa revelou que as empresas gostariam de experimentar o Modelo de Proteção de Software por Certificação Digital. As empresas alegaram ser um fator positivo o fato de poderem ter mais controle sobre as cópias de programas vendidas e de poderem anular as cópias piratas, revogando os certificados de licença. Quanto a possibilidade de controlar a concessão de software pela validade do certificado de licença, uma das empresas acreditou não ser uma boa decisão, visto que os usuários não iriam concordar.

- **Conclusão:**

  O resultado final obtido com a pesquisa feita para este grupo de empresas é bastante motivador. As empresas se mostraram aceitáveis ao novo Modelo de Proteção proposto e revelaram estarem dispostas a experimenta-lo. A pesquisa revelou também que algumas formas de proteção relatadas no item 2.3.3 foram empregadas como forma de combate à pirataria pelas empresas, embora sem resolver o problema com eficácia desejável.

A.3. Questionário Aplicado para Ensaio Preliminar de um Estudo de Campo

Questões de aspecto geral:

1. Para você, a pirataria ou cópias não autorizadas de programas para computador é uma atividade:
   ( ) que a maioria de usuários pratica com freqüência pois não há punição;
   ( ) normal e praticada porque não há nenhum mal nisto;
   ( ) normal e praticada porque não há meios eficientes que a impeçam;
   ( ) que não há como evitar devido ao preço dos produtos.

2. Você já possuiu ou copiou programas piratas?
   ( ) Sim ( ) Não

3. Na sua opinião, qual é a principal fonte de obtenção de programas pirata?
   ( ) Internet;
   ( ) Fornecedores de "pacotaços" de cds pela Internet, jornais e revistas;
   ( ) Camelódromo e lojas suspeitas;
   ( ) Amigos que emprestam para copiar.

4. Você já encontrou ou soube de alguém que teve que comprar um programa porque ninguém conseguiu copiar ou não foi achado em nenhum site da Internet?
   ( ) Sim
   ( ) Não
   ( ) Sim, mas é só ter paciência que logo sai uma versão pirata.

5. Se os preços dos programas fossem reduzidos drasticamente, você passaria a comprar cópias originais mesmo se fosse possível obtê-las de graça, ainda que de modo ilegal pela Internet?
   ( ) Sim, compraria ou já compro;
   ( ) Não, faria download de graça.

6. Você já ouviu falar de algum desses assuntos: certificado digital, assinatura digital, autoridade certificadora, chave privada, não repúdio?
   ( ) Sim, conheço todos eles;
   ( ) Mais ou menos. Alguns sim, outros não;
   ( ) Não, nenhum.

7. Digamos que você tenha experimentado um software útil para você, mas que o tempo de demonstração tenha expirado. Você está disposto a legalizar a situação e para isto precisa adquirir um certificado digital que vai seguramente, habilitar todas as funções do programa, permitindo a você instalar e desinstalar o software em qualquer computador que você queira, desde que siga os termos de compromisso do software. Você vê algum problema ou empecilho nisto?
   ( ) Não, por mim tudo bem, desde que o software funcione perfeitamente;
   ( ) Sim.

8. Se lhe disser que para piratear o software acima você teria que fornecer junto com o software o seu certificado digital, sua chave privada e que com isto você poderia ser processado por crime de pirataria, sem ter como negar o crime. Mediante isto, a sua opinião mudaria?
   ( ) Não permanece a mesma, mas não emprestaria/forneceria meu certificado e chave privada pra ninguém;
   ( ) Sim, mudaria.

Questões de aspecto empresarial:

1. Em quais ramos sua empresa se encaixa melhor?
   ( ) Prestadora de pequenos serviços em informática;
   ( ) Produtora de software;
   ( ) Criação e desenvolvimento WEB;
   ( ) Provedor de acesso;
   ( ) Suporte e treinamento;
   ( ) Revenda de produtos.

2. Sua empresa vivencia ou já vivenciou problemas com a pirataria de software, tendo sofrido prejuízos com isto?
   ( ) Sim ( ) Não

3. Sua empresa já tentou soluções para evitar a pirataria de software?
   ( ) Sim. Quais?
   ( ) Número de série;
   ( ) Disquete de proteção;
   ( ) Contra-senha;
   ( ) Dispositivo de hardware.
   ( ) Não

4. Atualmente sua empresa trabalha com algum tipo de proteção que seja 100% satisfatório?
   ( ) Sim. Qual?
   ( ) Não

5. Na sua opinião, a Proteção de Software por Certificação Digital seria aplicável na sua empresa?
   ( ) Não, há muitos elementos envolvidos, teria que contratar pessoal especializado e é muito complicado;
   ( ) Sim, arriscaria uma chance para ver ser funciona.

6. Na sua opinião, o fato de ter maior controle sobre as cópias de programas piratas e com a possibilidade de poder neutralizá-las de um momento para outro é um fator positivo para sua empresa?

   ( ) Sim, claro. Com isto posso controlar mais a pirataria, punir os responsáveis e evitar maiores prejuízos;

   ( ) Não acredito que isto vá funcionar.

7. O que você acha da possibilidade de não mais vender cópias de programas, mas sim conceder o direito de uso delas por um período estabelecido em contrato, como se fosse um aluguel?

   ( ) Excelente, pois além de baixar o preço final do produto e aumentar sua saída, poderia me gerar uma receita mais prolongada. Com isto, poderia contratar mais pessoal e reinvestir no software;

   ( ) Não, acho que os consumidores não aceitariam;

   ( ) Não, creio que o controle de expiração do certificado não iria funcionar.